Fondamenti del linguaggio LaTeX per la scrittura scientifica

Lezione 3: Gestione di documenti complessi

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

Dipartimento di Matematica, Politecnico di Milano

28 Novembre 2022

Questioni di impaginazione

Questioni di impaginazione, parte 1: allineamento

Gli ammbienti `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 1: allineamento

Gli ammbienti `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 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 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.

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 `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 3: 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 4: 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, in rari casi, un po' di compromesso.

Questioni di impaginazione, parte 4: 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, in rari casi, un po' di compromesso.

Questioni di impaginazione, parte 4: 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, in rari casi, un po' di compromesso.

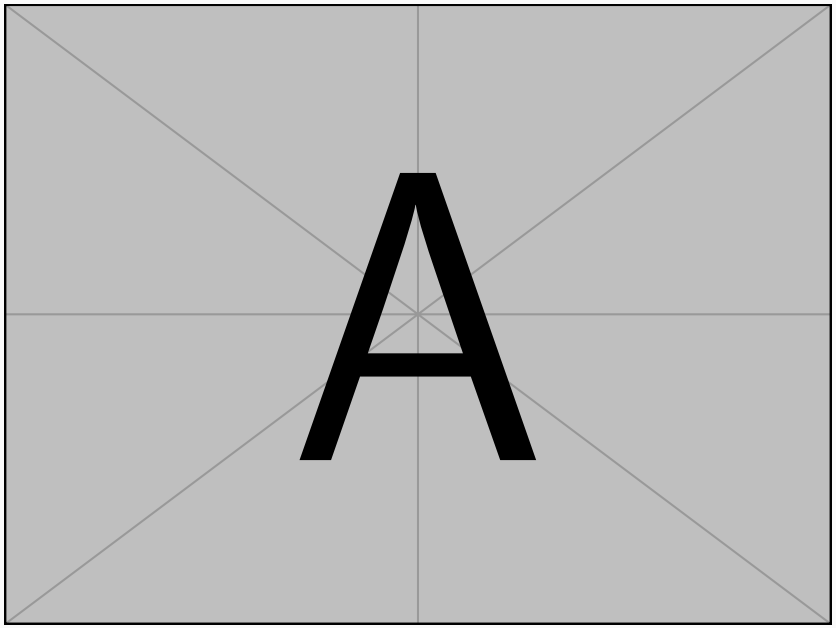
Questioni di impaginazione, parte 4: qualche avvertimento

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, in rari casi, un po' di compromesso.

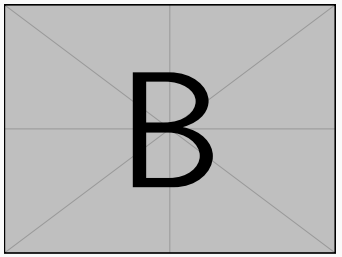
Esercizio 1 (ripresa): l'occhio vuole la sua parte!

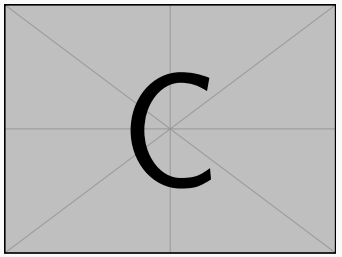
Replicare il seguente output:

La prima lettera dell'alfabeto è illustrata nella Figura seguente. Appare, nella lingua italiana, con una frequenza del 12% circa (fonte Wikipedia).



Altre lettere includono la B e la C, meno frequenti, illustrate di seguito.





La tabella riassume le loro frequenze alla seconda cifra decimale:

A	11.74%
B	0.92%
C	4.50%

Motivazione: oltre brevi articoli

Oltre la classe `article`, parte 1: le classi `report` e `book`

Le classi di documento `report` e `book` vengono utilizzate per documenti generalmente lunghi come tesi o libri.

Importante!

Nonostante esistano comandi *class-specific*, tutto quello che abbiamo visto fin adesso può essere utilizzato indipendentemente dalla `documentclass` scelta. Tuttavia, alcune direttive, come `\maketitle`, possono avere output diversi (rispetto a quanto visto per la classe `article`).

Oltre la classe `article`, parte 1: le classi `report` e `book`

Le classi di documento `report` e `book` vengono utilizzate per documenti generalmente lunghi come tesi o libri.

Importante!

Nonostante esistano comandi *class-specific*, tutto quello che abbiamo visto fin adesso può essere utilizzato indipendentemente dalla `documentclass` scelta. Tuttavia, alcune direttive, come `\maketitle`, possono avere output diversi (rispetto a quanto visto per la classe `article`).

Oltre la classe `article`, parte 2: classi personalizzate

Invece di usare le classi di default, può capitare di utilizzare classi personalizzate create da terzi. Esse sono codificate in file `.cls`, e per utilizzarli è sufficiente includerli nella cartella di lavoro, chiamando la documentclass corretta.

Ad esempio, è stato recentemente pubblicato il template LaTeX ufficiale della Scuola 3I.

Importante!

Quando si usano classi personalizzate, viene di solito fornito un template `.tex`. Utilizzare sempre quello come punto di partenza, in quanto tiene conto di questioni tecniche che l'utente finale non deve essere tenuto a gestire.

Oltre la classe `article`, parte 2: classi personalizzate

Invece di usare le classi di default, può capitare di utilizzare classi personalizzate create da terzi. Esse sono codificate in file `.cls`, e per utilizzarli è sufficiente includerli nella cartella di lavoro, chiamando la documentclass corretta.

Ad esempio, è stato recentemente pubblicato il template LaTeX ufficiale della Scuola 3I.

Importante!

Quando si usano classi personalizzate, viene di solito fornito un template `.tex`. Utilizzare sempre quello come punto di partenza, in quanto tiene conto di questioni tecniche che l'utente finale non deve essere tenuto a gestire.

Oltre la classe `article`, parte 2: classi personalizzate

Invece di usare le classi di default, può capitare di utilizzare classi personalizzate create da terzi. Esse sono codificate in file `.cls`, e per utilizzarli è sufficiente includerli nella cartella di lavoro, chiamando la documentclass corretta.

Ad esempio, è stato recentemente pubblicato il template LaTeX ufficiale della Scuola 3I.

Importante!

Quando si usano classi personalizzate, viene di solito fornito un template `.tex`. Utilizzare sempre quello come punto di partenza, in quanto tiene conto di questioni tecniche che l'utente finale non deve essere tenuto a gestire.

Oltre la classe `article`, parte 3: documenti complessi

Per "documento complesso" si intende un elaborato di lunghezza medio-lunga, spesso strutturato in capitoli ed implementato tramite la classe `report` o `book` (o una classe personalizzata simile).

A causa della loro lunghezza, si usa dividere il codice in più file.

Warning!

Per il resto della lezione, useremo come "prototipo" di documento complesso una tesi di laurea, e come `documentclass` la classe `book`. Tuttavia, larga parte delle considerazioni sono applicabili ad altri documenti complessi come libri o lunghi report scientifici.

Cominciamo a programmare!

Create **una nuova cartella** e nominatela "Lezione 3". Al suo interno, create `Lezione3.tex`:

```
\documentclass[10pt, a4paper]{book} % N.B.!
\usepackage[italian]{babel}
\usepackage{amsmath, amssymb, amsthm}
\usepackage[pass]{geometry} % N.B.!
% inserire nome, titolo e data (a piacere)
\begin{document}
        \maketitle
\end{document}
```

Struttura di un documento complesso

Si usa dividere il contenuto di un documento complesso in tre parti:

- *front matter*: frontespizio, dichiarazione di copyright (per libri), abstract, indice, lista di figure e tabelle; numerazione romana;
- *main matter*: il contenuto effettivo dell'elaborato; numerazione araba;
- *back matter*: appendici e bibliografia; numerazione araba.

La classe `book`, parte 1: impaginazione

Con le opzioni `oneside` e `twoside` possiamo indicare a LaTeX di modificare la spaziatura dei margini in modo da lasciare spazio per la rilegatura:

```
\documentclass[10pt, a4paper, oneside]{book} %
    singola facciata, sempre a destra
\documentclass[10pt, a4paper, twoside]{book} %
    fronte-retro
```

L'opzione di default è `twoside`. Ogni nuova "parte" del documento inizia automaticamente da una pagina destra (lasciando una pagina vuota se necessario).

La classe `book`, parte 1: impaginazione

Con le opzioni `oneside` e `twoside` possiamo indicare a LaTeX di modificare la spaziatura dei margini in modo da lasciare spazio per la rilegatura:

```
\documentclass[10pt, a4paper, oneside]{book} %
    singola facciata, sempre a destra
\documentclass[10pt, a4paper, twoside]{book} %
    fronte-retro
```

L'opzione di default è `twoside`. Ogni nuova "parte" del documento inizia automaticamente da una pagina destra (lasciando una pagina vuota se necessario).

La classe `book`, parte 2: front, main e back matter

I comandi `\frontmatter`, `\mainmatter` e `\backmatter` gestiscono le tre parti dell'elaborato. La struttura del codice sarà dunque come segue:

```
\documentclass[10pt, a4paper, twoside]{book}
% inclusione di pacchetti e completamento
    preambolo
\begin{document}
        \maketitle
        \frontmatter
        % contenuto front matter
        \mainmatter
        % contenuto main matter
        \backmatter
        % contenuto back matter
\end{document}
```

La classe `book`, parte 2: front, main e back matter

I comandi `\frontmatter`, `\mainmatter` e `\backmatter` gestiscono le tre parti dell'elaborato. La struttura del codice sarà dunque come segue:

```
\documentclass[10pt, a4paper, twoside]{book}
% inclusione di pacchetti e completamento
    preambolo
\begin{document}
        \maketitle
        \frontmatter
        % contenuto front matter
        \mainmatter
        % contenuto main matter
        \backmatter
        % contenuto back matter
\end{document}
```

La classe book, parte 3: frontespizio

Nella classe book, un'intera pagina è dedicata al frontespizio. Di default, essa viene trattata come una pagina qualunque, dunque viene numerata e, se si usa l'opzione twoside, viene anche spostata opportunamente. Una semplice soluzione è questa:

```
\usepackage[pass]{geometry} % includo il
   pacchetto senza modificare i margini
\begin{document}
  \begin{titlepage}
        \newgeometry{margin=2cm} % sistema
           margini localmente
        \thispagestyle{empty} % no numbering
        \maketitle
  \end{titlepage}
\end{document}
```

Strutturare un documento di classe `report` o `book`

Ricordiamo che, per strutturare un documento di classe `report` o `book`, è possibile utilizzare i comandi:

- `\chapter`, `\chapter*`,
- `\section`, `\section*`,
- `\subsection`, `\subsection*`,
- `\subsubsection`, `\subsubsection*` (usare solo se necessario),

in questo ordine gerarchico.

Front matter, parte 1: contenuti minimi

Generalmente, anche se non è convenzione universale, la front matter di una tesi contiene, nell'ordine:

- Abstract (un capitolo non numerato);
- Sommario (la traduzione italiana dell'Abstract, capitolo non numerato);
- Ringraziamenti (un capitolo non numerato);
- Indice dei contenuti della tesi;
- Lista di figure e tabelle (se presenti).

Front matter, parte 2: implementazione

L'implementazione della front matter è molto semplice, poiché LaTeX gestisce dinamicamente tutto per noi.

```
\begin{document}
        % ...codice titlepage...
        \frontmatter
        \chapter*{Abstract}
        Abstract in inglese.
        \chapter*{Sommario}
        Sommario in italiano.
        \chapter*{Ringraziamenti}
        Ringraziamenti.
        \tableofcontents
        \listoffigures
        \listoftables
\end{document}
```

Main matter, parte 1: inserimento di contenuti

La main matter è, essenzialmente, una successione di capitoli.

```
\begin{document}
        % ...codice frontmatter...
        \mainmatter
        \chapter{Capitolo 1}
        % Contenuti capitolo 1...
        \chapter{Capitolo 2}
        % Contenuti capitolo 2...
\end{document}
```

Tuttavia, in questo modo, il codice può diventare lungo e poco leggibile!

Main matter, parte 2: i comandi `input` e `include`

È uso comune quello di scrivere ogni capitolo in un file `.tex` dedicato. Il comando

```
\input{filename}
```

inserisce i contenuti del file nel punto in cui il comando viene chiamato.

```
% Lezione3.tex
\begin{document}
  % frontmatter...
  \mainmatter
  \input{chapter1.tex}
  \input{chapter2.tex}
\end{document}
```

```
% chapter1.tex
\chapter{Capitolo 1}
% Contenuti capitolo 1...
```

```
% chapter2.tex
\chapter{Capitolo 2}
% Contenuti capitolo 2...
```

Main matter, parte 2: i comandi `input` e `include`

Il comando `\include{filename}` fa la stessa cosa, a livello macroscopico. Tuttavia, ammette la specifica nel preambolo

```
% nel preambolo
\includeonly{file1, file2, ...}
```

che esclude automaticamente i file non menzionati nella lista di argomenti. Ciò lo rende preferibile per progetti lunghi ed elaborati per alleggerire i tempi di compilazione e aiutare la manutenzione del codice.

Digressione: cartella di lavoro

Importante!

Mantenete organizzata la cartella di lavoro! Ad esempio, create la sottocartella `chapters` e inserite dentro i file `.tex` dei singoli capitoli. Per comunicare a LaTeX dove cercare i file, inserite il percorso come argomenti di `input` o `include`, ad esempio inserire `\input{chapters/chapter1.tex}`.

Digressione: progetti con più file `.tex`

Quando si spezza il codice LaTeX in più file, non tutti sono compilabili. Ad esempio, ad alcuni file manca il preambolo, e se proviamo a compilarli otteniamo solo un messaggio di errore.

Infatti, in teoria, dovremmo sempre compilare *solo* il file completo di preambolo da cui stiamo chiamando, ad esempio, `\input`.

Tuttavia, possiamo dire a LaTeX: "quando tento di compilare un singolo capitolo, compila automaticamente il file corretto".

Digressione: progetti con più file `.tex`

```
% chapters/chapter1.tex
%! TeX root = ../filename.tex
\chapter{Capitolo importato da un altro file}
% contenuti del capitolo...
```

Compilando `chapter1.tex` sarà invece compilato `filename.tex` (`../` vuol dire "cerca il file nella cartella superiore a questa").

Main matter, parte 3: contenuto dei capitoli

Il contenuto dei capitoli può essere redatto con i comandi che abbiamo visto nelle precedenti lezioni: testo, matematica, floats (ed anche altro!).

Main matter, parte 4: labeling e referencing

Tutto ciò che è numerato (capitoli, sezioni, teoremi, floats, equazioni, ecc...) può essere referenziato in modo dinamico.

Esempio: se scrivo l'equazione

$$a^2 + b^2 = c^2, \tag{1}$$

posso inserire nel testo un riferimento all'Equazione (1).

Main matter, parte 4: labeling e referencing

Ad ogni elemento numerato è possibile associare un'etichetta, che è poi possibile referenziare nel testo.

```
\label{keyword} % creare un'etichetta
\ref{label} % comando generale per riferirsi ad
   un'etichetta
\eqref{label} % specifica per le equazioni
```

Main matter, parte 4: labeling e referencing

```
Esempio: se scrivo l'equazione
\begin{equation} \label{eq:equazione1}
        a^2+b^2 = c^2,
\end{equation}
posso inserire nel testo un riferimento all'
   Equazione \eqref{eq:equazione1}.
```

Per mantenere il codice ordinato, scrivere le label subito dopo o subito prima di ciò che si vuole numerare, e di utilizzare delle keyword del tipo cosa:nome, ad esempio `eq:maxwell`, in modo da navigare più comodamente quando il numero delle label cresce.

Main matter, parte 4: labeling e referencing

Esempi di labeling:

```
\chapter{Capitolo 1} \label{ch:cap1} % label di
    un capitolo
\begin{equation} \label{eq:pitagora}
        a^2+b^2 = c^2
\end{equation}
\begin{figure}
        \includegraphics{filename}
        \label{fig:figura1}
        \caption{contenuto...}
\end{figure}
Nel capitolo \ref{ch:cap} vi sono l'equazione
\eqref{eq:equazione1} e la Figura
\ref{fig:figura1}.
```

Main matter, parte 4: labeling e referencing

Importante!

Negli ambienti per float, inserire le label *sempre* prima della caption.

Back matter, parte 1: contenuti minimi

La back matter solitamente contiene:

- una o più Appendici;
- Bibliografia.

Tuttavia, LaTeX non considera le appendici propriamente back matter, e ha definito una sorta di "appendix matter" con il comando `\appendix`.

Back matter, parte 2: appendici

Ogni capitolo dopo \appendix è considerato parte dell'appendice, verrà "numerato" con lettere a posto di numeri, e chiamato Appendice anziché Capitolo.

```
\begin{document}
        %  main matter...
        \appendix
        \chapter{Prima appendice} % od anche
            \input{chapters/appendixA.tex}
        % altre appendici...
        \backmatter
        % resto della back matter...
\end{document}
```

Back matter, parte 3: bibliografia

Per implementare la bibliografia, necessitiamo del pacchetto `biblatex`. Una possibile inclusione è

```
% nel preambolo
\usepackage[style=numeric, sorting=nty, backend=
    bibtex]{biblatex}
```

Back matter, parte 3: bibliografia

Per implementare la bibliografia seguiremo questo procedimento:

1. Inseriamo la lista di fonti (libri, articoli, ecc...) in un file separato con il formato `.bib`;
2. Nel preambolo indichiamo a LaTeX di cercare le fonti in questo file `.bib`;
3. Nel corpo inseriamo riferimenti e bibliografia.

Back matter, parte 3: bibliografia

Per implementare la bibliografia seguiremo questo procedimento:

1. Inseriamo la lista di fonti (libri, articoli, ecc...) in un file separato con il formato `.bib`;
2. Nel preambolo indichiamo a LaTeX di cercare le fonti in questo file `.bib`;
3. Nel corpo inseriamo riferimenti e bibliografia.

Back matter, parte 3: bibliografia

Per implementare la bibliografia seguiremo questo procedimento:

1. Inseriamo la lista di fonti (libri, articoli, ecc...) in un file separato con il formato `.bib`;
2. Nel preambolo indichiamo a LaTeX di cercare le fonti in questo file `.bib`;
3. Nel corpo inseriamo riferimenti e bibliografia.

Bibliografia, parte 1: file `.bib`

Creiamo un nuovo file su TEXStudio e salviamolo come `sources.bib`. All'interno andranno tutte le referenze, non necessariamente ordinate. Il file contiene una successione di strutture di questo tipo:

```
% sources.bib
@what{keyword,
        tag = {value1},
        tag2 = {value2},
        ...
}
```

Bibliografia, parte 1: file `.bib`

```
% sources.bib
@article{testart21,
        author = {Surname1, Name1 and Surname2,
            Name2 and Surname3, Name3},
        title = {Test Article},
        year = {2021},
        pages = {1-100},
        journal = {J. Ver. Import. Thing.},
        volume = {1},
}
```

Bibliografia, parte 1: file `.bib`

```
% sources.bib
@book{testbook21,
        author = {Surname1, Name1 and Surname2,
            Name2 and Surname3, Name3},
        title = {Test Article},
        year = {2021},
        series = {Lectures in Very Important
            Things},
        publisher = {Springer},
        volume = {1},
}
```

Bibliografia, parte 2: specificare il file `.bib`

Nel preambolo del file root, inseriamo, dopo aver inserito il pacchetto `biblatex`,

```
% nel preambolo
\addbibresource{sources.bib}
```

Bibliografia, parte 3: citare una fonte nel testo

Nel corpo del documento, è possibile citare una fonte con il comando `\cite{keyword}`:

```
% nel corpo
In questo lavoro,
   applicheremo un
   metodo presentato
   in \cite{testart
   21}.
```

```
% sources.bib
@article{testart21,
         ...
}
```

In questo lavoro, applicheremo un metodo presentato in [1].

Bibliografia, parte 4: stampare la bibliografia

Con il comando

```
\printbibliography[heading=bibintoc, title=
    Bibliografia]
```

stampiamo la bibliografia con il titolo specificato. L'opzione heading inserisce la Bibliografia nell'Indice.

Bibliografia, parte 4: stampare la bibliografia

```
\begin{document}
        % main matter, appendici
        \backmatter
        \printbibliography[heading=bibintoc,
           title=Bibliografia]
\end{document}
```

Comandi e pacchetti personalizzati

Direttive personalizzate, parte 1: `newcommand`

È possibile inserire, nel preambolo, comandi ed ambienti personalizzati. Molto spesso, è utile definirsi delle semplici macro. Il comando

```
% nel preambolo
\newcommand{\name}[nargs]{def}
```

definisce il comando `\name`. Esso richiede `n` argomenti obbligatori, da indicarsi tra graffe come usuale. Nella definizione, si chiama un argomento utilizzando `#x` con `x` intero tra 1 ed `n`.

Direttive personalizzate, parte 1: `newcommand`

Ad esempio, si può ridefinire l'integrale

$$\int_a^b f(x)\,\mathrm{d}x$$

tramite

```
\newcommand{\myint}[4]{\int_{#1}^{#2} #3 \: \
    mathrm{d}#4}
\begin{document}
        \begin{equation*}
                \myint{a}{b}{f(x)}{x}
        \end{equation*}
\end{document}
```

Direttive personalizzate, parte 2: `newenvironment`

Creare ambienti è possibile secondo la stessa logica, anche se è un po' più laborioso.

```
% nel preambolo
\newenvironment{name}[nargs]{begindef}{enddef}
% Idea dell'implementazione:
% codice in begindef
% contenuto dell'ambiente
% codice in enddef

% Il contenuto dell'ambiente NON si considera un
    argomento!
```

Direttive personalizzate, parte 2: `newenvironment`

Ad esempio, creo un ambiente atto a contenere un esercizio in un box.

```
\newenvironment{esercizio}[1][Esercizio]{\vspace
    {\baselineskip} \begin{tabular}{|p{\textwidth
    }|} \hline\textbf{#1}. }{ \\ \hline\end{
    tabular} \vspace{\baselineskip}}
% nome del nuovo ambiente: esercizio
% argomenti: 1, con valore di default Esercizio
% begin: lascia una riga vuota, poi imposta una
    tabella con una sola colonna larga quanto
    possibile, scrivendo in grassetto il primo
    argomento come intestazione
% end: chiude la riga e la "scatola", chiude la
    tabella e lascia un'altra riga vuota.
```

Direttive personalizzate, parte 2: `newenvironment`

Ad esempio, creo un ambiente atto a contenere un esercizio in un box.

```
\newenvironment{esercizio}[1][Esercizio]{\vspace
   {\baselineskip} \begin{tabular}{|p{\textwidth
   }|} \hline\textbf{#1}. }{ \\ \hline\end{
   tabular} \vspace{\baselineskip}}
\begin{document}
        \begin{esercizio}[Esercizio piu'
           difficile]
                Calcolare i valori interi
                   positivi di n per cui n!
                   termina con esattamente 1000
                   zeri.
        \end{esercizio}
\end{document}
```

Direttive personalizzate, parte 3: pacchetti

Per conservare un gruppo di comandi e/o ambienti in modo da poterli utilizzare in più progetti, conviene fare un pacchetto. I pacchetti sono codificati in file di stile `.sty`.

File di stile (cenni): costruire un pacchetto

```
% mypackage.sty
```

File di stile (cenni): costruire un pacchetto

```
% mypackage.sty
\NeedsTeXFormat{LaTeX2e}
```

File di stile (cenni): costruire un pacchetto

```
% mypackage.sty
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{mypackage}[2021/12/09 My
    Package]
\RequirePackage{packagename} % dipendenze da
    altri pacchetti
```

File di stile (cenni): costruire un pacchetto

```
% mypackage.sty
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{mypackage}[2021/12/09 My
    Package]
\RequirePackage{packagename} % dipendenze da
    altri pacchetti
% tutti i comandi e gli ambienti...
\endinput % fine del file
```

Salvate il pacchetto nella cartella di lavoro e potrà essere incluso nel `.tex` con `\usepackage{mypackage}`.

Grazie dell'attenzione!

Prossima lezione: 05/12/2022 in ?.?.?