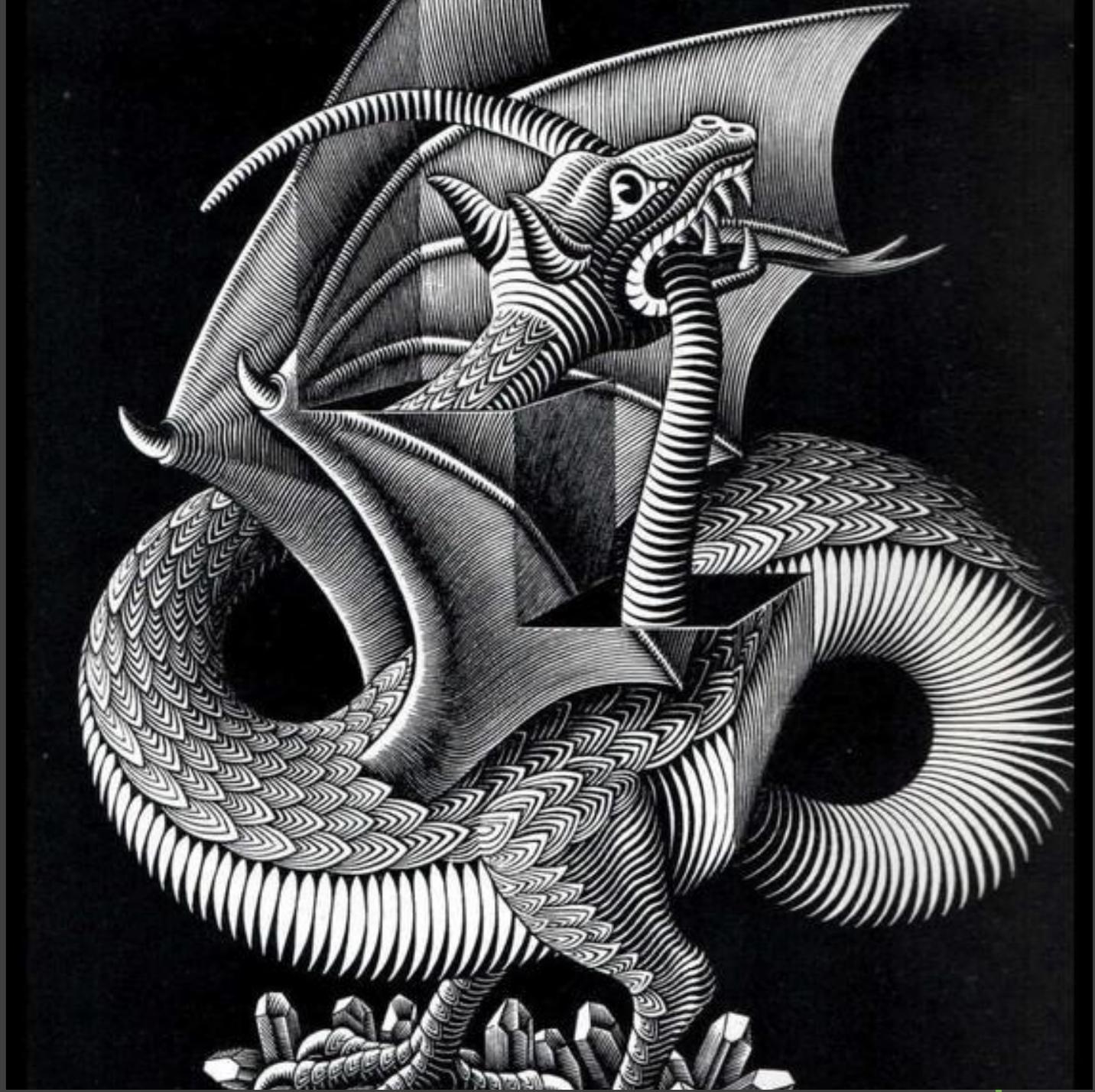


Introduzione a LaTeX

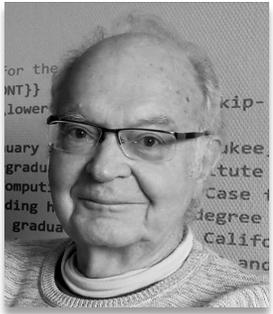


Linguaggi di mark-up

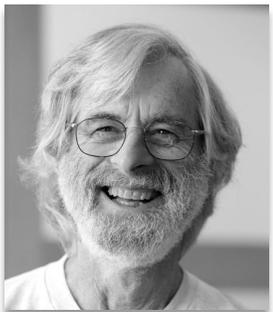
- Un *linguaggio di mark-up* è un linguaggio artificiale progettato per **arricchire un testo con dei meta-dati** aggiuntivi che ne caratterizzano l'**organizzazione logica** o lo **stile tipografico** in modo da poterne produrre un *rendering* di qualità adatta allo strumento con cui si intende fruire del documento stesso (pagina web, documento stampato, documento in formato digitale, PDF, ecc.)
- Sono linguaggi di mark-up:
 - **HTML**, *HyperText Mark-up Language*: il linguaggio di mark-up per la definizione di pagine web ipertestuali, definito da **Tim Berners Lee**, insieme al protocollo HTTP (Hyper Text Transfer Protocol)
 - **TeX**: il linguaggio di mark-up per la produzione di testi scientifici di qualità tipografica, progettato da **Donald Knuth**; il principale punto di attenzione è lo stile tipografico del testo e la rappresentazione di espressioni matematiche
 - **LaTeX**: il linguaggio di mark-up costruito su TeX da **Leslie Lamport**, per semplificare la redazione di documenti complessi; oltre alle caratteristiche di TeX, il principale punto di attenzione è nella definizione di una struttura logica del documento, basata su un «modello» di base che ne caratterizza anche l'impostazione tipografica generale
- Questi linguaggi sono derivati da **SGML**, *Standard Generalized Mark-up Language*, con cui possono essere definite le «grammatiche» di altri linguaggi di mark-up
- Oggi è largamente diffuso, per l'organizzazione logica di dati sulla base di un formato aperto, il linguaggio **XML**, *eXtensible Mark-up Language*



Tim Berners Lee
(1955)



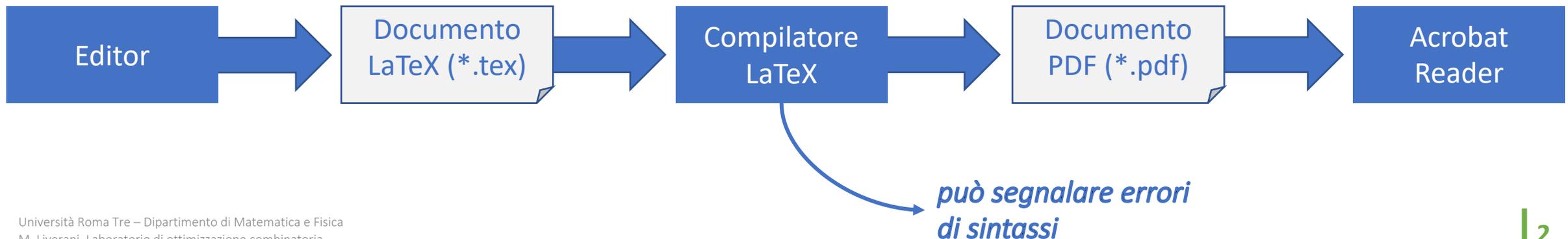
Donald Knuth
(1938)



Leslie Lamport
(1941)

Linguaggi di mark-up e rendering del documento

- Un linguaggio di mark-up è un linguaggio artificiale, ma non è un linguaggio di programmazione: non serve a codificare algoritmi
- I linguaggi di mark-up si usano integrando con degli specifici **tag** (comandi) un file di testo in cui è riportato il contenuto del documento
- Il **file sorgente** arricchito con i tag di mark-up è comunque un documento di testo che è facilmente comprensibile da chi conosce il linguaggio di mark-up utilizzato, ma non è ancora il documento finale
- Per produrre il documento finale occorre elaborare il documento contenente i tag del linguaggio di mark-up con un **motore di rendering** (un programma in grado di interpretare il linguaggio di mark-up) che possa così produrre il file nella sua versione finale
 - HTML → web browser che interpretando il documento in formato HTML visualizza una pagina web
 - TeX / LaTeX → compilatore che interpreta i linguaggi TeX o LaTeX che produce un file in formato DVI (*device independent*) o PDF (*portable document format*)



Tipi di documento in LaTeX

LaTeX consente di organizzare il contenuto del documento sulla base di diversi schemi standard (altri se ne possono aggiungere con pacchetti aggiuntivi), chiamati **document class**:

- **Lettere** (*letter*): sono dei documenti privi di una struttura particolare, se non un'intestazione con i riferimenti del mittente e del destinatario della lettera
- **Articoli** (*article*): sono documenti di più pagine, ma non molto lunghi, organizzati in sezioni; ciascuna sezione può essere organizzata in più sotto-sezioni; si conclude con dei riferimenti bibliografici
- **Report** (*report*): è un documento più lungo, con una copertina, un indice degli argomenti, organizzato in capitoli, suddivisi in sezioni e sotto-sezioni; è impaginato tipicamente per essere stampato solo sul «fronte» delle pagine e non anche sul retro; si conclude con dei riferimenti bibliografici
- **Libri** (*book*): è un documento lungo, simile ad un report, con una copertina, un insieme di parti preliminari (indice degli argomenti, introduzione, ecc.), il cui corpo principale è organizzato in capitoli suddivisi in sezioni e sotto-sezioni; è seguito da eventuali appendici, da un indice analitico e dai riferimenti bibliografici; i libri sono impaginati per essere stampati fronte/retro, con i capitoli che iniziano sempre da una pagina dispari
- **Slide** (*slide*): sono documenti scritti con caratteri molto grandi, senza un'organizzazione specifica dei contenuti, utili per essere proiettati su un grande schermo nel corso di una conferenza o di un seminario

Struttura di un articolo

Documento LaTeX (documentclass: **article**)

Intestazione
(titolo, autore, data,
inclusione di altri
package)

Corpo del documento
Titolo, autore, data (*make title*)

Sintesi / Sommario (abstract)
Testo della sintesi preliminare

Sezione (section)
Testo della sezione

Sotto-sezione (subsection)
Testo della sotto-sezione

...

Bibliografia (bibliography)
Elenco delle fonti bibliografiche

Struttura di un report o di un libro

Documento LaTeX (documentclass: **book**)

Intestazione
(titolo, autore, data,
inclusione di altri
package)

Corpo del documento

«front matter»

Copertina (`\maketitle`)

Indice degli argomenti
(table of contents)

Introduzione

«main matter»

Capitolo (`\chapter`)

Sezione (`\section`)

Sotto-sezione
(`\subsection`)

Capitolo (`\chapter`)

Sezione (`\section`)

Sotto-sezione
(`\subsection`)

Capitolo (`\chapter`)

Sezione (`\section`)

Sotto-sezione
(`\subsection`)

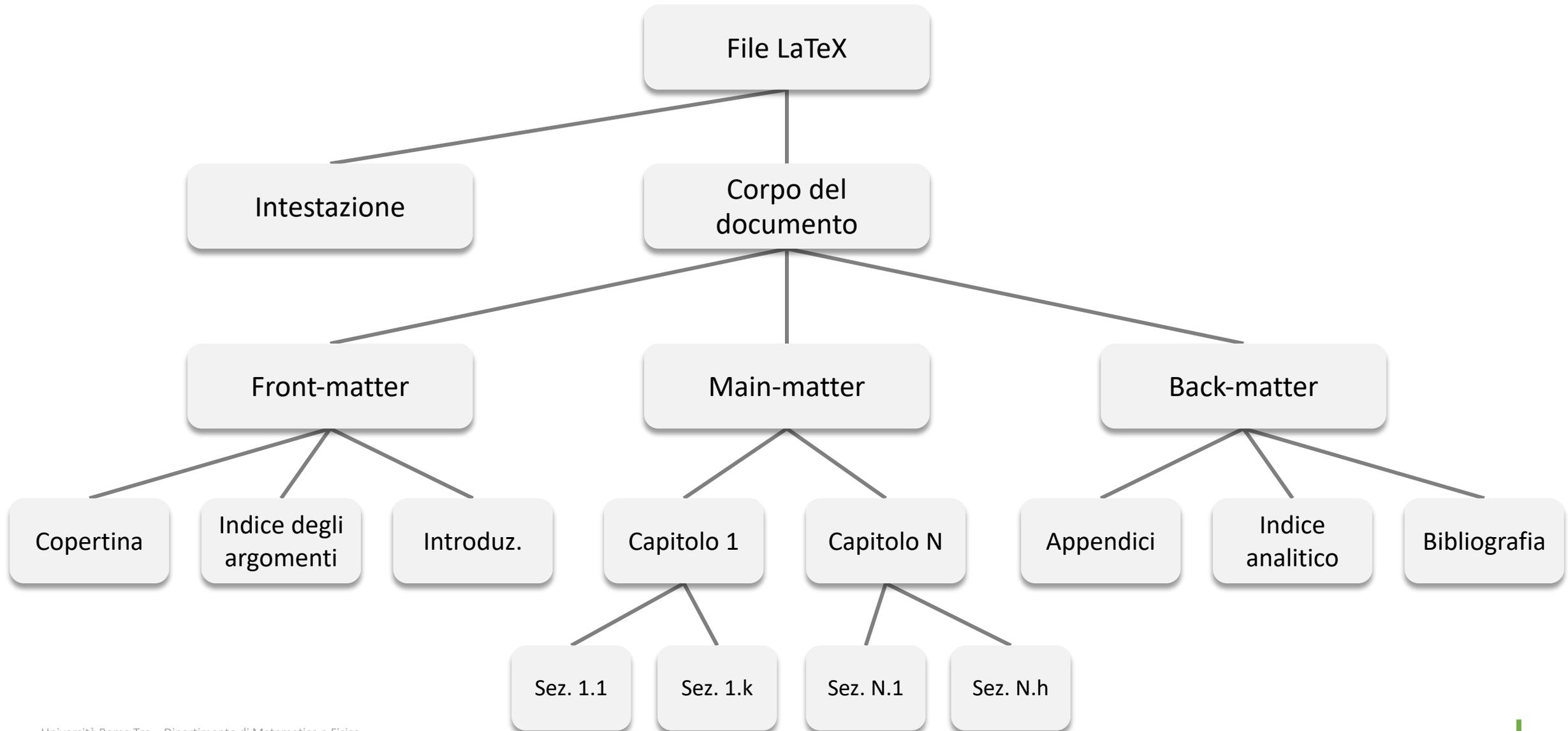
«back matter»

Appendici

Indice analitico

Bibliografia

Struttura di un report o di un libro



Sintassi generale

- I tag o comandi di LaTeX possono essere di tre tipi:
 - comandi «singoli», a sé stanti, tipicamente utili per definire un carattere particolare
 - comandi che si applicano ad un blocco di testo
 - comandi che definiscono un «ambiente» (*environment*) al cui interno possono essere eseguiti comandi che al di fuori di tale ambiente non possono essere usati; i comandi che definiscono degli ambienti possono prevedere dei parametri
- Tutti i comandi LaTeX sono costituiti da una parola chiave (senza spazi, case sensitive), preceduta da un backslash «\»
- I blocchi di testo sono delimitati da parentesi graffe «{...}», che possono essere pure nidificate più volte
- i comandi che definiscono un ambiente iniziano con un meta-comando «\begin[*parametri*]{*ambiente*}» e terminano con il meta-comando «\end{*ambiente*}»
- Visto lo scopo del LaTeX ci sono due ambienti speciali che servono per riportare espressioni matematiche che, siccome sono molto usate, hanno un simbolo compatto per definirli:
 - $\$$ *espressione matematica* $\$$ in linea in un paragrafo di testo
 - $\$\$$ *espressione matematica* $\$\$$ centrata su una riga a sé stante

Un semplice esempio

```
\documentclass{article}
```

tipo di documento

```
\usepackage[italian]{babel}
```

sillabazione in italiano
e traduzione delle
etichette

```
\usepackage[utf8]{inputenc}
```

pacchetti aggiuntivi

```
\usepackage[T1]{fontenc}
```

sintesi preliminare
dell'articolo

```
\title{Un semplice esempio}
```

titolo prima sezione

```
\author{Marco Liverani}
```

```
\date{12 aprile 2021}
```

titolo, data e autore del
documento

```
\begin{document}
```

qui inizia il documento vero e proprio

```
\maketitle
```

qui scriviamo il titolo, l'autore e la
data definiti prima

```
\begin{abstract}
```

```
In questo breve articolo parlerò di ...
```

```
\end{abstract}
```

```
\section{Antefatto}
```

C'era una volta, tanti, ma tanti anni fa,
in un castello sperduto tra le montagne...

riga vuota per far
andare a capo il testo

Nel castello viveva una bella principessa
di nome Ermenegilda che amava la
matematica: $f(x) = \frac{\sin x}{x}$

una formula inserita
nel testo

```
\section{Epilogo}
```

```
... e vissero tutti felici e contenti
```

```
\end{document}
```

qui termina il documento (quel che
segue non viene elaborato da LaTeX)

Alcuni comandi di base per la formattazione del testo

- `\chapter{titolo del capitolo}`: intestazione del capitolo (solo per documenti di tipo «book» e «report»)
- `\section{titolo della sezione}`: intestazione della sezione
- `\subsection{titolo della sotto-sezione}`: intestazione della sotto-sezione
- `\subsubsection{titolo della sotto-sotto-sezione}`: intestazione della sotto-sotto-sezione (sconsigliato)
- `\paragraph{titolo}`: titolo di un paragrafo (non si usa quasi mai)
- `\textbf{testo in grassetto}`: rappresenta il testo in grassetto (*bold face*)
- `\textsf{testo in carattere senza grazie}`: rappresenta il testo con il carattere di default senza grazie (*sans serif*)
- `\emph{testo evidenziato}`: rappresenta il testo in corsivo (se il paragrafo non è in corsivo)
- `\texttt{testo a spaziatura fissa}`: rappresenta il testo con il carattere a spaziatura fissa (teletype)
- `\verb'testo non interpretato'`: riporta il testo così come è scritto, con carattere a spaziatura fissa
- `\begin{verbatim}testo non interpretato\end{verbatim}`: riporta il testo così come è scritto, con carattere a spaziatura fissa, anche su più righe

Liste ed elenchi

■ Elenchi puntati:

- `\begin{itemize}`: inizia un elenco puntato
- `\item testo dell'elenco`: voce dell'elenco puntato (ripetuta più volte, per ogni voce dell'elenco)
- `\end{itemize}`: termina l'elenco puntato

■ Elenchi numerati:

- `\begin{enumerate}`: inizia un elenco numerato
- `\item testo dell'elenco`: voce dell'elenco numerato (ripetuta più volte, per ogni voce dell'elenco)
- `\end{enumerate}`: termina l'elenco numerato

■ Elenchi con descrizione:

- `\begin{description}`: inizia un elenco con voci descritte
- `\item[voce dell'elenco] testo descrittivo della voce dell'elenco`: voce dell'elenco (ripetuta più volte, per ogni voce dell'elenco)
- `\end{description}`: termina l'elenco descrittivo

Figure e note

- È possibile inserire nel testo una figura presente su un file estero (tipicamente un file in formato PDF, PNG o JPEG) includendo nell'intestazione del documento il package **Graphicx** con il comando

```
\usepackage{graphicx}:
```

```
\begin{figure}
```

```
\begin{center}
```

```
\includegraphics[scale=0.5]{nomefile.pdf}
```

```
\caption{didascalia della figura}
```

```
\label{etichetta per successivi riferimenti alla figura}
```

```
\end{center}
```

```
\end{figure}
```

- La figura viene rappresentata in una pagina in cui LaTeX ritiene corretto posizionarla, sulla base delle regole canoniche dell'impaginazione tipografica
- È possibile riferirsi ad una figura con il comando `\ref` o `\pageref`:
«si veda la Figura `\ref{etichetta}` riportata a pagina `\pageref{etichetta}` ...»
- Note a piè di pagina con il comando `\footnote``{nota a piè di pagina}`:
«Un ordinamento topologico `\footnote{Un ordinamento topologico è un ordinamento parziale ...}` lo si ottiene con una visita in profondità...»

Pseudo-codice

- I package *Algorithmic* e *Algorithm* mettono a disposizione rispettivamente dei comandi e un ambiente per la rappresentazione dello pseudo-codice di un algoritmo; possono essere inclusi con i seguenti comandi:

```
\usepackage{algorithmic}
```

```
\usepackage[chapter,boxed]{algorithm}
```

- I comandi per lo pseudo codice sono i seguenti (da riportare con lettere maiuscole):

- `\STATE`: istruzione generica, es.: un'assegnazione
- `\IF{condizione}`: istruzione condizionale
- `\ELSE`: alternativa in un'istruzione condizionale
- `\ENDIF`: chiusura di un'istruzione condizionale
- `\FOR{condizione}`, `\FORALL{condizione}`, `\ENDFOR`: struttura iterativa
- `\WHILE{condizione}`, `\ENDWHILE{condizione}`: struttura iterativa
- `\REPEAT`, `\UNTIL{condizione}`: struttura iterativa
- `\REQUIRE{input}`: descrizione dell'input per l'algoritmo
- `\ENSURE{output}`: descrizione di ciò che l'algoritmo restituisce in output

Pseudo-codice

```
\begin{algorithm}
  \caption{BFS( $G$ ,  $s$ )}
  \begin{algorithmic}[1]
    \REQUIRE{Il grafo  $G$  ed una sorgente  $s \in V(G)$ }
    \ENSURE{I vertici visitati su  $G$  a partire da  $s$ }
    \STATE  $Q = \{s\}$ ,  $T = (\{s\}, \emptyset)$ 
    \STATE  $D(v) = \infty$  per ogni  $v \in V(G)$ ,  $D(s) = 0$ 
    \STATE marca  $s$  di grigio e tutti gli altri di bianco
    \WHILE  $\{Q \neq \emptyset\}$ 
      \STATE estrai un elemento  $v$  da  $Q$ 
      \FORALL  $\{u \in N(v)\}$ 
        \IF  $\{u \text{ non è marcato}\}$ 
          \STATE marca  $u$  di grigio, aggiungi  $u$  a  $Q$ 
          \STATE  $D(u) = D(v) + 1$ 
          \STATE aggiungi  $u$  e  $(v,u)$  all'albero  $T$ 
        \ENDIF
      \ENDFOR
      \STATE marca  $v$  di nero
    \ENDWHILE
  \end{algorithmic}
\end{algorithm}
```

Algoritmo 11 BFS(G , s)

Input: Il grafo G ed una sorgente $s \in V(G)$

Output: I vertici visitati su G a partire da s

1: $Q = \{s\}$, $T = (\{s\}, \emptyset)$

2: $D(v) = \infty$ per ogni $v \in V(G)$, $D(s) = 0$

3: marca s di grigio e tutti gli altri di bianco

4: **fintanto che $Q \neq \emptyset$ ripeti**

5: estrai un elemento v da Q

6: **per ogni $u \in N(v)$ ripeti**

7: **se u non è marcato allora**

8: marca u di grigio, aggiungi u a Q

9: $D(u) = D(v) + 1$

10: aggiungi u e (v,u) all'albero T

11: **fine-condizione**

12: **fine-ciclo**

13: marca v di nero

14: **fine-ciclo**

Codice di linguaggio di programmazione

- È possibile includere nel testo del documento LaTeX un codice sorgente in un linguaggio di programmazione standard (C, Java, Python, SQL, ecc.) che il compilatore rappresenterà correttamente indentato ed evidenziando le istruzioni del programma
- Occorre includere il package *Listings*: `\usepackage{listings}`
- Il package può essere configurato con le seguenti istruzioni da riportare nell'intestazione del documento:

```
\lstset{
  language=python,
  basicstyle=\scriptsize, showstringspaces=false, basicstyle=\sffamily,
  frameround=tttt, frame=single, columns=fullflexible,
  tabsize=3, numbers=left, stepnumber=1, numberstyle=\tiny\sffamily
}
```

- Per includere un file sorgente esterno:
- Per riportare un codice direttamente nel documento:

```
\lstinputlisting{nomefile.py}
```

```
\begin{lstlistings}
codice sorgente del programma
\end{lstlistings}
```

Bibliografia

- Per i riferimenti bibliografici si può usare l'ambiente «thebibliography»:

```
\begin{thebibliography}{99}
```

```
\bibitem{etichetta} Nome Cognome, \emph{titolo articolo}, Titolo rivista, Editore, Anno, numero
```

```
\bibitem{etichetta} Nome Cognome, \emph{titolo libro}, Editore, edizione, Anno
```

```
\end{thebibliography}
```

- Esempio:

```
\begin{thebibliography}{99}
```

```
\bibitem{Aho1} A.V. Aho, J.E. Hopcroft, J.D. Ullman, \emph{Data structures and algorithms}, Addison-Wesley, 1987.
```

```
\bibitem{Aho2} A.V. Aho, J.E. Hopcroft, J.D. Ullman, \emph{The Design and Analysis of Computer Algorithms}, Addison-Wesley, 1974.
```

```
\bibitem{Rivest} T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, \emph{Introduzione agli algoritmi e strutture dati}, terza edizione, McGraw-Hill, 2010.
```

```
\bibitem{Liverani1} M. Liverani, \emph{Programmare in C -- Guida al linguaggio attraverso esercizi svolti e commentati}, seconda edizione, Società Editrice Esculapio, 2013.
```

```
\bibitem{Tarjan} Robert Endre Tarjan, \emph{Data Structures and Network Algorithms}, SIAM, 1983.
```

```
\end{thebibliography}
```

- I riferimenti nel testo si fanno con `\cite{etichetta}`

Espressioni matematiche

- I seguenti comandi valgono solo all'interno di un ambiente matematico: $\$ \dots \$$ oppure $\$\$ \dots \$\$$
- Nel testo i numeri o i semplici nomi di variabili vanno sempre riportati dentro ad un ambiente matematico; ad esempio:

«Sia $\$G = (V,E)\$$ un grafo con $\$n\$$ vertici ed $\$m\$$ spigoli»

Espressione	Simbolo	Esempio	Risultato
Apici / potenze	<code>^</code>	<code>x^2</code>	x^2
Pedici	<code>_</code>	<code>x_i</code>	x_i
Apici e pedici lunghi	<code>^{...}</code> / <code>_{...}</code>	<code>x_{i,j}^{(k)}</code>	$x_{i,j}^{(k)}$
Radici	<code>\sqrt</code> / <code>\root</code> <code>\of</code>	<code>\sqrt{x+2} + \root 3 \of 2</code>	$\sqrt{x+2} + \sqrt[3]{2}$
Confronti	<code><</code> <code>\leq</code> <code>=</code> <code>\not=</code> <code>></code> <code>\geq</code>	<code>a < b \leq c = d \not= e > f \geq g</code>	$a < b \leq c = d \neq e > f \geq g$
Insiemi	<code>\in</code> <code>\not\in</code> <code>\cup</code> <code>\cap</code> <code>\subset</code> <code>\subseteq</code>	<code>x \in A, y \not\in B, A \cup B = C, A \cap B = \emptyset, A \subset C, B \subseteq C</code>	$x \in A, y \notin B, A \cup B = C, A \cap B = \emptyset, A \subset C, B \subseteq C$
Simboli	<code>\infty</code> <code>\pi</code> <code>\alpha</code> <code>\times</code> <code>\to</code> <code>\forall</code> <code>\exists</code>	<code>\infty \pi \alpha \times \to \forall \exists</code>	$\infty \pi \alpha \times \rightarrow \forall \exists$

Espressioni matematiche

Espressione	Simbolo	Esempio	Risultato
Limiti	<code>\lim</code>	<code>\lim_{x \to \infty} f(x)</code>	$\lim_{x \rightarrow \infty} f(x)$
Sommatorie	<code>\sum</code>	<code>\sum_{i=1}^n w(v_i)</code>	$\sum_{i=1}^n w(v_i)$
Prodotti	<code>\prod</code>	<code>\prod_{k=1}^n k = n!</code>	$\prod_{k=1}^n k = n!$
Integrali	<code>\int</code>	<code>\int_{-k}^k f(x) dx</code>	$\int_{-k}^k f(x) dx$
Frazioni	<code>\frac{...}{...}</code>	<code>\frac{n(n+1)}{2}</code>	$\frac{n(n+1)}{2}$
Parentesi	<code>\left\{ \left(\right) \right\}</code>	<code>\left\{ \left((x+y)^k \right) \right\}</code>	$\left\{ \left((x+y)^k \right) \right\}$
Funzioni	<code>\sin \cos \log \max \min</code>	<code>\sin x + \cos x - \log_2 n + \max_{i < k} x_i + \min_{j=1, \dots, n} x_j</code>	$\sin x + \cos x - \log_2 n + \max_{i < k} x_i + \min_{j=1, \dots, n} x_j$