

# Introduzione a $\text{\LaTeX}$

---

Perché  $\text{\LaTeX}$ ?

Struttura di un documento  $\text{\LaTeX}$

Formattazione di un documento

Espressioni matematiche

Tabelle

Immagini

Ambienti matematici

Riferimenti e citazioni

Perché  $\text{\LaTeX}$ ?

---

In ambito scientifico la grande maggioranza delle pubblicazioni è scritta utilizzando  $\text{\LaTeX}$ , siano esse articoli, report o libri.

$\text{\LaTeX}$  è un formattatore di testo, quindi per ottenere il documento desiderato si deve scrivere un file contenente la lista di istruzioni necessarie a crearlo.

Per visualizzare il risultato, il file contenente i comandi (estensione `.tex`) deve essere compilato ed eseguito. Al termine di questa operazione si ottiene un documento `.pdf`.

L<sup>A</sup>T<sub>E</sub>X è utilizzato per la sua versatilità nell'editing di testi scientifici contenenti simboli matematici, tabelle e diagrammi.

*Esempio di diagramma fatto con L<sup>A</sup>T<sub>E</sub>X (Snake Lemma)*

$$\begin{array}{ccccccc}
 & & & c_q \in C_q & \xrightarrow{g_q} & z''_q \in C''_q & \longrightarrow 0 \\
 & & & \downarrow \partial_q & & \downarrow \partial''_q & \\
 0 \longrightarrow & c'_{q-1} \in C'_{q-1} & \xrightarrow{f_{q-1}} & \partial_q(c_q) \in C_{q-1} & \xrightarrow{g_{q-1}} & 0 \in C''_{q-1} & \longrightarrow 0 \\
 & \downarrow \partial'_{q-1} & & & & & \\
 & 0 \in C_{q-2} & & & & & 
 \end{array}$$

## Come lo utilizzeremo noi

Per scrivere documenti in  $\text{\LaTeX}$  utilizzeremo Overleaf, un  $\text{\LaTeX}$  editor disponibile online, perciò ad ogni studente è richiesto di creare un account.

Per prendere confidenza con  $\text{\LaTeX}$  e non disperdere quanto appreso si consiglia in futuro di scrivere le relazioni relative agli esperimenti svolti in una qualsiasi disciplina (e.g. fisica, scienze) con questo strumento, previo consenso da parte del docente. Verrà fornito un template da cui è possibile prendere spunto per le prime volte.

# Struttura di un documento $\text{\LaTeX}$

---

# Struttura di un documento $\text{\LaTeX}$

```
\documentclass[opzioni]{classe}

% Preambolo
\usepackage[utf8]{inputenc}
\usepackage[italian]{babel}
...

% Documento
\begin{document}
Testo che si desidera stampare
...
\end{document}
```



# Struttura di un documento $\text{\LaTeX}$

Ogni documento  $\text{\LaTeX}$  inizia con il comando

```
\documentclass[opzioni]{classe}
```

Tra le classi più comuni ci sono:

- Article [noi useremo questa]
- Report/Book
- Beamer

Tra le opzioni più comuni ci sono:

- grandezza dei caratteri misurata in punti (e.g. 12pt)
- dimensione del foglio (e.g. a4paper)

Noi useremo quasi sempre

```
\documentclass[12pt, a4paper]{article}
```

# Struttura di un documento $\text{\LaTeX}$

Il preambolo è la parte compresa tra  
`\documentclass[opzioni]{classe}` e `begin{document}`.

In questo spazio si inseriscono in genere:

- pacchetti da caricare
- opzioni generali (a volte richieste dalla classe del documento)
- personalizzazioni di comandi e ambienti

Questi due pacchetti permettono al programma di riconoscere la tastiera italiana e utilizzare la corretta sillabazione perciò vanno aggiunti nel preambolo.

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

# Formattazione di un documento

---

L<sup>A</sup>T<sub>E</sub>X gestisce in maniera autonoma la formattazione del testo, andando a capo e gestendo i paragrafi in base ai comandi specificati.

Ogni blocco di testo viene interpretato come un paragrafo e all'inizio di ogni paragrafo c'è un rientro. Per creare un nuovo paragrafo si lascia una riga vuota tra i due blocchi di testo oppure il comando `\par`.

Per andare a capo senza rientro si può utilizzare il comando `\\` oppure `\linebreak` se si vuole giustificare la riga. Il comando `\noindent` sopprime il rientro ad inizio del paragrafo successivo.

# Personalizzare il Font

Per scegliere il tipo di font si carica nel preambolo il pacchetto desiderato (e.g. `\usepackage{palatino}`).

Una lista con i font disponibili si può trovare al seguente link [https://www.overleaf.com/learn/latex/font\\_typefaces](https://www.overleaf.com/learn/latex/font_typefaces)

Per modificare lo stile del font si utilizzano i comandi:

- `\textbf{...}` per il **grassetto** [Ctrl+B]
- `\textit{...}` per il *corsivo* [Ctrl+I]
- `\underline{...}` per il sottolineato
- `{\em ...}` dipende dal *contesto*

# Personalizzare il Font

La dimensione base del font è definita nel comando iniziale

```
\documentclass[11pt, ...]{...}.
```

Per modificare la dimensione del font di una porzione di testo si utilizza `\begin{grandezza}...\end{grandezza}`.

Le grandezze più comuni sono `tiny` `small` `Large` `huge`

Per specificare tipo di font e grandezza in punti si può usare il seguente comando:

```
{\fontsize{size}{interlinea}\fontfamily{font}\selectfont ...}
```

Esempio:

```
{\fontsize{10}{12}\fontfamily{ptm}\selectfont testo...}
```

da come output: testo...

*Nota Bene:* la classe *article* supporta come grandezze 10pt, 11pt, 12pt. Per utilizzare una grandezza differente bisogna importare uno specifico pacchetto.

Per modificare il colore si importa il pacchetto

```
\usepackage[dvipsnames]{xcolor}
```

e si utilizzano i comandi:

- `{\color{nome_colore} ... }` **cambia il colore del testo**
- `\colorbox{nomecolore}{...}` **evidenzia il testo**



In  $\text{\LaTeX}$  il testo è giustificato di default

Si può allineare il testo a sinistra, centro, destra con i seguenti comandi:

- `\begin{flushleft} ... \end{flushleft}`
- `\begin{center} ... \end{center}`
- `\begin{flushright} ... \end{flushright}`

Per fare liste numerate si utilizza

```
\begin{enumerate}  
  \item ...  
  \item ...  
\end{enumerate}
```

Esempio:

1. primo elemento
2. secondo elemento
3. terzo elemento

Per fare liste non numerate si utilizza

```
\begin{itemize}  
  \item ...  
  \item ...  
\end{itemize}
```

Esempio:

- primo elemento
- secondo elemento
- terzo elemento

Per rendere il file `.tex` più leggibile è utile inserire delle righe di commento che spiegano la funzione dei comandi successivi.

Queste linee sono introdotte dal simbolo `%` che dice al programma di ignorarle quando il documento viene compilato, perciò la loro presenza non ha nessuna influenza sul `.pdf` finale.

Per stampare un comando si può utilizzare `\verb|...|` che disabilita la sua esecuzione. Ad esempio, per stampare il comando `\LaTeX` devo scrivere `\verb|\LaTeX|` altrimenti otterrò  $\text{\LaTeX}$ .

Per inserire stampare più linee di codice in un file .tex basta racchiuderlo tra i comandi `\begin{verbatim}` ...  
`\end{verbatim}`.

```
\documentclass[12pt, a4paper]{article}

% Pacchetti per la lingua
\usepackage[utf8]{inputenc}
\usepackage[italian]{babel}

\begin{document}
    Questo è il mio primo documento con \LaTeX.
\end{document}
```

L'ambiente *verbatim* permette di stampare le stringhe dei comandi  $\text{\LaTeX}$ , tuttavia se si vuole inserire un algoritmo in pseudocodice un pacchetto utile sono *algorithm* e *algpseudocode*.

Nel preambolo andranno inseriti i due import `\usepackage{algorithm}` e `\usepackage{algpseudocode}` e poi sarà possibile utilizzare l'ambiente *algorithm*.

```
\begin{algorithm}[H]
  \caption{}
  \begin{algorithmic}[1]
    \State ...
    \State ...
  \end{algorithmic}
\end{algorithm}
```

```
\begin{algorithm}[H]
\caption{Calcolo del fattoriale}
\begin{algorithmic}[1]
  \State \textbf{Input:}  $N$ 
  \State  $f_{\text{fattoriale}} \leftarrow 1$ 
  \For{ $i = 1$  \textbf{to}  $N$ }
    \State  $f_{\text{fattoriale}} \leftarrow f_{\text{fattoriale}} \cdot i$ 
  \EndFor
  \State \textbf{Return}  $f_{\text{fattoriale}}$ 
\end{algorithmic}
\end{algorithm}
```

---

**Algorithm 1** Calcolo del fattoriale

---

```
1: Input:  $N$   
2:  $fattoriale \leftarrow 1$   
3: for  $i = 1$  to  $N$  do  
4:    $fattoriale \leftarrow fattoriale \cdot i$   
5: end for  
6: Return  $fattoriale$ 
```

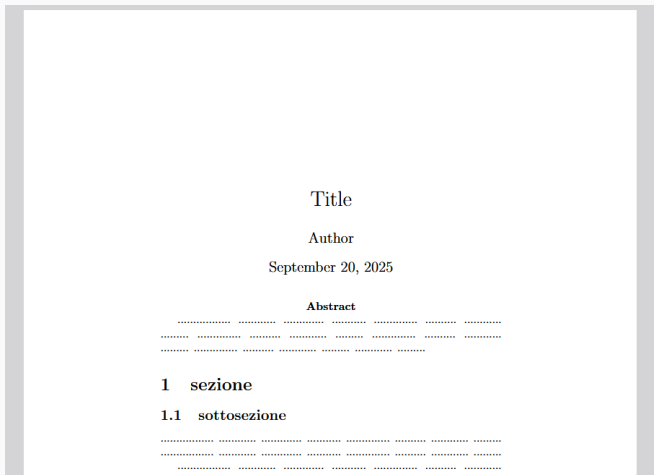
---



# Esempio classe Lettera



**Figure 1:** Esempio di lettera in  $\text{\LaTeX}$



**Figure 2:** Esempio di articolo in  $\text{\LaTeX}$

# Espressioni matematiche

---

I pacchetti più utilizzati per scrivere in ambiente matematico sono `{amsmath}` e `{amssymb}` e vanno importati nel preambolo

```
\usepackage{amsmath}
```

```
\usepackage{amssymb}
```

L<sup>A</sup>T<sub>E</sub>X ha due modi matematici, detti *inline* e *display*. Il primo serve per formule posizionate all'interno del testo, il secondo per formule che sono poste su una riga separata.

In entrambi i modi l'espressione matematica deve essere racchiusa tra due delimitatori.

Per il modo *inline* i delimitatori sono `\(...\)`, `$...$`, `\begin{math}...\end{math}` che producono tutti lo stesso risultato.

Es: Due variabili  $x$  e  $y$  si dicono *inversamente proporzionali* se il loro prodotto  $xy$  è uguale a una costante  $k \neq 0$ .

Per il modo *display* i delimitatori sono `\[ ... \]` e `\begin{equation} ... \end{equation}` che producono tutti lo stesso risultato.

Spesso funziona anche `$$ ... $$` ma è considerato obsoleto.

Es: Dato il polinomio

$$3x^2 - 4x + 3 \tag{1}$$

determinare, se esistono, le radici reali.

I caratteri inseriti in un ambiente matematico sono considerati parte di un'unica espressione e la spaziatura viene ignorata:

`\[f(x)\neq 0\forall x \in X \]`

$$f(x) \neq 0 \forall x \in X$$

La spaziatura può essere inserita per esempio con i comandi `\` e `\quad`. Es: `\[f(x)\neq 0 \quad \forall x \in X \]`

$$f(x) \neq 0 \quad \forall x \in X$$

Per cambiare il font in modo matematico di solito si utilizza:

- `\mathbf` ... per il grassetto (es: sia  $\mathbf{x}$  un vettore)
- `\mathbb` ... per blackboard bold (es:  $\forall x \in \mathbb{R}$ )
- `\mathcal` ... per il calligrafico ( $\mathcal{G}$ )

ma ci sono molte altre opzioni ...



## Simboli comuni

$\rightarrow$	<code>\rightarrow</code> o <code>\to</code>	$\leftarrow$	<code>\leftarrow</code> o <code>\gets</code>
$\Longleftarrow$	<code>\Longleftarrow</code>	$\Longrightarrow$	<code>\Longrightarrow</code>
$\Leftrightarrow$	<code>\Leftrightarrow</code>	$\Longleftrightarrow$	<code>\Longleftrightarrow</code>
$\pm$	<code>\pm</code>	$\times$	<code>\times</code>
$\in$	<code>\in</code>	$\subset$	<code>\subset</code>
$\subseteq$	<code>\subseteq</code>	$\subsetneq$	<code>\subsetneq</code>
$\exists$	<code>\exists</code>	$\forall$	<code>\forall</code>
$\sim$	<code>\sim</code>	$\geq$	<code>\geq</code>
$=$	<code>=</code>	$\neq$	<code>\neq</code>
$\equiv$	<code>\equiv</code>	$\simeq$	<code>\simeq</code>
$<$	<code>&lt;</code>	$>$	<code>&gt;</code>
$\leq$	<code>\leq</code>	$\geq$	<code>\geq</code>
$\cup$	<code>\cup</code>	$\cap$	<code>\cap</code>

# Alfabeto greco

$\alpha$	<code>\alpha</code>	$\iota$	<code>\iotaota</code>	$\varrho$	<code>\varrhorho</code>
$\beta$	<code>\betaeta</code>	$\kappa$	<code>\kappaappa</code>	$\sigma$	<code>\sigmaigma</code>
$\gamma$	<code>\gammaamma</code>	$\lambda$	<code>\lambdabda</code>	$\varsigma$	<code>\varsigmaigma</code>
$\delta$	<code>\deltaelta</code>	$\mu$	<code>\muu</code>	$\tau$	<code>\tauau</code>
$\epsilon$	<code>\epsilonpsilon</code>	$\nu$	<code>\nuu</code>	$\upsilon$	<code>\upsilonpsilon</code>
$\varepsilon$	<code>\varepsilonpsilon</code>	$\xi$	<code>\xii</code>	$\phi$	<code>\phii</code>
$\zeta$	<code>\zetaeta</code>	$\omicron$	<code>o</code>	$\varphi$	<code>\varphiphi</code>
$\eta$	<code>\etaeta</code>	$\pi$	<code>\pii</code>	$\chi$	<code>\chihi</code>
$\theta$	<code>\thetaeta</code>	$\varpi$	<code>\varpii</code>	$\psi$	<code>\psii</code>
$\vartheta$	<code>\varthetaeta</code>	$\rho$	<code>\rhoi</code>	$\omega$	<code>\omegai</code>
$\Gamma$	<code>\Gammaamma</code>	$\Xi$	<code>\Xii</code>	$\Phi$	<code>\Phii</code>
$\Delta$	<code>\Deltaelta</code>	$\Pi$	<code>\Pi i</code>	$\Psi$	<code>\Psi i</code>
$\Theta$	<code>\Theta eta</code>	$\Sigma$	<code>\Sigma i</code>	$\Omega$	<code>\Omega i</code>
$\Lambda$	<code>\Lambda mbda</code>	$\Upsilon$	<code>\Upsilon i</code>		

## Apici e pedici

Per aggiungere apici e pedici si usano rispettivamente `_` e `^`.

$$\begin{array}{l|l} \$x_{\{0\}}\$ & x_0 \\ \$2^{\{x\}}\$ & 2^x \end{array}$$

Apici e pedici sono mostrati in maniera differente in base a se si utilizza il modo *inline* o *display*.

$$\sum_{i=0}^n i = \frac{n(n+1)}{2} \quad \int_0^{10} f(x) \quad \lim_{x \rightarrow +\infty} \frac{1}{x}$$

$$\sum_{i=0}^n i = \frac{n(n+1)}{2} \quad \int_0^{10} f(x) \quad \lim_{x \rightarrow +\infty} \frac{1}{x}$$

Altri delimitatori comuni sono:

- (...) parentesi tonde (comando `$(...)$`)
- [...] parentesi quadre (comando `$[...]$`)
- {...} parentesi graffe (comando `$\{...\}$`)
- ||...|| valore assoluto

per adattare le dimensioni si aggiunge `\left` e `\right` davanti alle rispettive parentesi di apertura e chiusura.

`\[(\frac{1}{10})^x`:

$$\left(\frac{1}{10}\right)^x$$

`\[ \left(\frac{1}{10}\right)^x \]`:

$$\left(\frac{1}{10}\right)^x$$

Una matrice si scrive normalmente con il comando

```
\[  
\begin{matrix}  
a & b \\  
c & d  
\end{matrix}  
\]
```

e produce un risultato del tipo

$$\begin{matrix} a & b \\ c & d \end{matrix}$$

Inserendo *pmatrix*, *bmatrix*, *vmatrix* al posto di *matrix* nel comando precedente si ottiene rispettivamente:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad \begin{vmatrix} a & b \\ c & d \end{vmatrix}$$

Per scrivere sistemi si utilizza l'ambiente *cases* come in esempio

```
\[  
\begin{cases}  
  f(x)= x^2 \quad x \leq 0 \\  
  f(x)= \ln(x) \quad x > 0  
\end{cases}  
\]
```

produce

$$\begin{cases} f(x) = x^2 & x \leq 0 \\ f(x) = \ln(x) & x > 0 \end{cases}$$

## Allineamento formule

Per allineare le formule si può utilizzare l'ambiente *align* inserendo un `&` sui simboli da allineare.

Esempio:

```
\begin{align*}
210 &= 2\cdot 100+1\cdot 10+0\cdot 1\\
&= 2\cdot 10^2+1\cdot 10^1+0\cdot 10^0
\end{align*}
```

$$\begin{aligned} 210 &= 2 \cdot 100 + 1 \cdot 10 + 0 \cdot 1 \\ &= 2 \cdot 10^2 + 1 \cdot 10^1 + 0 \cdot 10^0 \end{aligned}$$



# Tabelle

---

## Tabelle: Tabular

L'ambiente *tabular* permette di creare tabelle in  $\text{\LaTeX}$  specificando il numero di colonne e l'allineamento

```
\begin{tabular}{l|c|r|r}  
  id & data & altezza & peso \\  
\hline  
$1234$ & 17/08/2012 & $176$ cm & $73$ kg \\  
$1235$ & 17/08/2012 & $178$ cm & $78$ kg \\  
$1236$ & 17/08/2012 & $182$ cm & $85$ kg \\  
\end{tabular}
```

In particolare, ci sarà una colonna per ogni *l*, *c*, *r* in

`\begin{tabular}{...}`, rispettivamente con allineamento a sinistra, centro e destra.

## Tabelle: Tabular

Gli elementi di ogni riga sono separati dal simbolo & e a fine riga va inserito `\\`. Il comando `\hline` permette di inserire una riga orizzontale mentre `|` si utilizza per inserire i bordi.

L'output del codice è il seguente:

id	data	altezza	peso
1234	17/08/2012	176 cm	73 kg
1235	17/08/2012	178 cm	78 kg
1236	17/08/2012	182 cm	85 kg

## Tabelle: Table

*tabular* è l'ambiente base per la creazione di tabelle in  $\text{\LaTeX}$ , tuttavia vi sono altri ambienti che permettono una maggiore personalizzazione.

Uno di questi è *table* che permette di aggiungere una didascalia e un tag per futuri riferimenti all'interno del testo

```
\begin{table}[h]
  \centering
  \begin{tabular}{c|c}
    & \\
    &
  \end{tabular}
  \caption{Caption}
  \label{tab:placeholder}
\end{table}
```

## Tabelle: NiceTabular

Un'altra opzione consiste nell'importare il pacchetto *nicematrix* nel preambolo [`\usepackage{nicematrix}`] e usare l'ambiente *NiceTabular*

```
\begin{NiceTabular}{ccc}  
  A & B & C \\  
  1 & 2 & 3 \\  
\end{NiceTabular}
```

## Tabelle: larghezza colonne

Per specificare la larghezza delle colonne si può utilizzare  $m\{larghezza\}$  al posto di l, c oppure r.

Per utilizzare questo comando è necessario importare il pacchetto *array* nel preambolo aggiungendo `\usepackage{array}`.

```
\begin{tabular}{m{1cm} | m{3cm} | m{3cm}}  
nome & cognome & età \\
```

## Tabelle: `resizebox`

A volte una tabella può risultare più grande dello spazio disponibile perciò può essere utile aggiustare manualmente la sua grandezza.

Per farlo si può utilizzare il comando `\resizebox{}{}{}` importando nel preambolo il pacchetto *graphicx* con `\usepackage{graphicx}`.

Nelle tre parentesi `\resizebox{1}{2}{3}` vanno rispettivamente inserite:

1. larghezza
2. altezza
3. contenuto

A volte desideriamo specificare solamente la larghezza oppure l'altezza mantenendo le proporzioni originali. Per farlo è sufficiente inserire la larghezza oppure l'altezza e nell'altro campo inserire un "!".

Esempio: `\resizebox{10cm}{!}{oggetto}`

La larghezza del testo può essere recuperata con il comando `\textwidth`.



## Tabelle: resizebox

```
\begin{table}[h]
  \centering
  \resizebox{0.4\textwidth}{!}{
    \begin{tabular}{l|c|r|r}
      id & data & altezza & peso \\
      \hline
      $1234$ & 17/08/2012 & $176$ cm & $73$ kg \\
      $1235$ & 17/08/2012 & $178$ cm & $78$ kg \\
      $1236$ & 17/08/2012 & $182$ cm & $85$ kg \\
    \end{tabular}
  }
  \caption{Caption}
  \label{tab:placeholder}
\end{table}
```

id	data	altezza	peso
1234	17/08/2012	176 cm	73 kg
1235	17/08/2012	178 cm	78 kg
1236	17/08/2012	182 cm	85 kg

**Table 1:** Caption

# Immagini

---

Nell'ambiente figure di solito si inseriscono:

- `\centering` per centrare la figura
- `includegraphics[grandezza]{img}` per includere l'immagine con una certa grandezza
- `\caption{Caption}` per inserire la didascalia
- `label{Label}` per riferimenti nel testo alla figura

Opzioni per definire la grandezza sono:

- `width=misura`
- `height=misura`
- `scale=fattore`



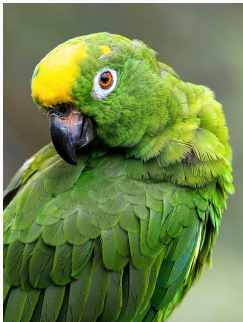
**Figure 3:** Margherita

La posizione dove l'immagine viene inserita è decisa autonomamente da  $\text{\LaTeX}$ , quindi può non corrispondere alla posizione in cui l'ambiente `figure` è inserito nel documento.

Se necessario è possibile forzare il comando aggiungendo l'opzione `[h]` all'ambiente `figure` come segue `\begin{figure}[h]`, tuttavia ciò non garantisce il risultato atteso.

A volte potremmo essere interessati ad inserire due immagini, una di fianco all'altro. Per fare ciò si può utilizzare l'ambiente *minipage* nel seguente modo:

```
\begin{figure}[h]
  \begin{minipage}{0.45\linewidth}
    \centering
    \includegraphics[width=...]{immagini/img_1.jpg}
    \caption{Pappagallo}
  \end{minipage}
  \begin{minipage}{0.45\linewidth}
    \centering
    \includegraphics[width=...]{immagini/img_2.jpg}
    \caption{Albatro}
  \end{minipage}
\end{figure}
```



**Figure 4:** Pappagallo



**Figure 5:** Albatro



# Ambienti matematici

---

$\text{\LaTeX}$  permette di inserire ambienti matematici come definizioni, postulati e teoremi. Per fare ciò si utilizza il comando `\newtheorem` che si utilizza nel seguente modo:

- Prima si inserisce `\newtheorem{nome}{testo}`
- Poi si inserisce l'enunciato compreso tra `\begin{nome} ... \end{nome}`

Per inserire un teorema si utilizza:

```
\newtheorem{tag}{Teorema}  
\begin{tag}  
...  
\end{tag}
```

Per inserire una dimostrazione invece si deve:

- aggiungere nel preambolo il pacchetto *amsthm* con  
`\usepackage{amsthm}`
- inserire la dimostrazione nel punto desiderato racchiusa tra  
`\begin{proof} ... \end{proof}`

Esempio:

```
\newtheorem{teo}{Teorema}
\begin{teo}
    La somma di due numeri pari è un numero pari.
\end{teo}
```

## **Teorema**

*La somma di due numeri pari è un numero pari.*

Esempio:

```
\begin{proof}
```

```
Siano  $a = 2m$  e  $b = 2n$  due numeri pari. \\
```

```
Allora  $a + b = 2m + 2n = 2(m + n)$ , che è pari.
```

```
\end{proof}
```

**Proof.**

Siano  $a = 2m$  e  $b = 2n$  due numeri pari.

Allora  $a + b = 2m + 2n = 2(m + n)$ , che è pari.



$\text{\LaTeX}$  permette di definire e personalizzare vari ambienti matematici per differenziare lo stile di definizioni, postulati, dimostrazioni, etc.

Noi non vedremo come si può fare, per il momento se volete definire un ambiente diverso, ad esempio *postulato*, potete utilizzare lo stesso comando cambiando il nome.

Esempio:

```
\newtheorem{post}{Postulato 1}  
\begin{post}  
    Due punti distinti determinano  
    una ed una sola retta a cui appartengono.  
\end{post}
```

## **Postulato 1**

*Due punti distinti determinano una ed una sola retta a cui appartengono.*

## Riferimenti e citazioni

---



In un file  $\text{\LaTeX}$  è possibile creare riferimenti ad ogni elemento numerato del testo, ad esempio a capitoli, sezioni, tabelle, equazioni e immagini.

Per fare ciò si assegna all'oggetto un'etichetta con il comando `\label{nome_etichetta}`.

Per aggiungere un'etichetta a un capitolo/sezione/sottosezione ad esempio basta scrivere:

```
\section{title} \label{sec:tag_sezione}
```

Per aggiungere un'etichetta ad una figura basta inserire nell'ambiente *figure* il comando `\label{fig:tag_figura}`.

Per far riferimento all'elemento desiderato nel testo è sufficiente utilizzare il comando `\ref{nome_etichetta}`.

Best practice: è possibile inserire un qualunque nome come etichetta, ma normalmente per semplificare la ricerca si mette un prefisso che indica il tipo di elemento (per esempio *fig:* per una figura) seguito da un nome che ci ricorda cosa rappresenta. Ad esempio, per una tabella contenente l'elenco degli studenti un'etichetta potrebbe essere *tab:studenti*.

Nel testo potrò poi richiamarla come in questo esempio: *L'elenco degli studenti è mostrato nella Tabella \ref{tab:studenti}*.

In ambito scientifico è sempre richiesto di specificare le fonti a cui si fa riferimento all'interno del testo.

Per fare questo normalmente si crea un file .bib in cui è inserito l'elenco delle fonti che sono poi importate nel file principale. Non vedremo come creare questo file, per il momento è sufficiente elencare le fonti alla fine del proprio report.

Per creare un collegamento a una pagina web si può importare nel preambolo il pacchetto *hyperref* con il comando `\usepackage{hyperref}`.

Per creare il link si utilizza il comando `\href{url}{link}`

Ad esempio:

`\href{https://www.guetti.tn.it/}{pagina_web_guetti}`  
crea il seguente collegamento `pagina_web_guetti`

Queste slide sono basate su un seminario tenuto dal prof Gianluca Occhetta per il corso di Matematica presso l'Università di Trento.

Il contenuto è stato riadattato per introdurre l'argomento a studenti delle scuole superiori.