

# Guida pratica agli esempi minimi compilabili

Ivan Valbusa  
ivan.valbusa@gmail.com

18 aprile 2021\*

## 1 Cos'è un esempio minimo compilabile

**Definizione** (Esempio minimo compilabile o EMC). Un EMC è un codice  $\text{\LaTeX}$  che si possa copiare sul proprio editor e compilare, per mostrare il problema, la difficoltà o l'errore lamentati dall'utente. Deve essere (1) *completo*, cioè deve contenere preambolo e corpo del documento, e (2) *minimo*, cioè contenere solo lo stretto necessario per riprodurre il problema.

Dalla definizione segue immediatamente il seguente

**Corollario.** Gli EMC permettono di isolare i problemi  $\text{\TeX}$ nici e perciò ne rendono più veloce ed efficace la risoluzione.

Un EMC adeguato per la domanda: «Come eliminare gli spazi tra i filetti verticali?»

```
\documentclass{article}
\usepackage{booktabs}

\begin{document}

\begin{tabular}{|c|c|c|c|}
\toprule
one & two & three & \\
one & two & three & \\
\bottomrule
\end{tabular}

\end{document}
```

Un EMC adeguato per la domanda: «Come ridurre lo spazio tra “L” e “orem”?»

```
\documentclass{book}
\usepackage{lettrine}
\usepackage{lipsum}

\begin{document}

\lettrine{L}{orem} ipsum dolor sit
amet. Cum sociis natoque penatibus
et magnis dis parturient montes,
nascetur ridiculus mus. Ut augue
sed urna pellentesque.

\end{document}
```

Come si può facilmente vedere, entrambi gli esempi contengono solo lo stretto indispensabile per riprodurre il problema in cerca di soluzione e sono pertanto *minimi*. Inoltre possono essere semplicemente copiati e compilati per mostrare il problema in oggetto, essendo *completi* di preambolo e corpo del documento. Di seguito vedremo degli esempi di codici che non rispondono a questi requisiti e che sono quindi non minimi o non compilabili.

---

\*Questo lavoro è rilasciato con licenza Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. Per leggere il testo della licenza, vai alla pagina: <http://creativecommons.org/licenses/by-nc-nd/4.0/>

## 2 Esempi non minimi e non compilabili

### 2.1 Esempi non minimi

I motivi principali per cui un codice risulta *non minimo* si possono ridurre alla presenza dei seguenti elementi nel file sorgente:

- pacchetti inutili;
- commenti superflui;
- contenuto sovrabbondante;
- codice formattato non correttamente.

Quello seguente è un codice *non minimo*, creato per rispondere alla domanda: «Perché la prima riga dopo il titolo di sezione non è rientrata?».

#### Esempio non minimo

```
1 \documentclass{book} % Classe per documenti con capitoli
2 \usepackage{lettrine}
3 \usepackage{lipsum}
4 \usepackage{array} % Per tabelle
5 \usepackage{booktabs} % Per linee tabelle migliori
6
7 \begin{document}
8 \section{
9 Lorem ipsum      }
10 \lipsum
11
12 Lorem ipsum dolor sit amet ipsum dolor sit amet ipsum
13 dolor sit amet ipsum dolor sit amet ipsum dolor sit amet
14 sit amet ipsum dolor sit amet ipsum dolor sit amet ipsum.
15 \end{document}
```

Come si vede facilmente, il codice contiene degli elementi perfettamente inutili ai fini della richiesta di aiuto. Innanzitutto carica pacchetti irrilevanti (lettrine, array, booktabs); alle righe 1, 3 e 4 ci sono commenti superflui; le righe 11-13 sono ugualmente inutili; infine, le righe 7-9 sono scritte malamente e rendono il codice poco chiaro.

### 2.2 Esempi non compilabili

I motivi principali per cui un codice risulta *non compilabile* si possono ridurre alla presenza dei seguenti elementi nel file sorgente:

- errori di sintassi;
- omissioni di codice essenziale;
- classi o pacchetti disponibili solo all'autore;
- comandi e/o ambienti non definiti nel codice;
- omissione di pacchetti necessari per compilare il codice;
- uso di immagini o altri file esterni disponibili solo all'autore.

Quello che segue è un codice *non compilabile* sebbene, in un certo senso, si possa considerare minimo.

#### Esempio non compilabile

```
1 \documentclass{myuniversityclass}
2 \usepackge{graphicx}
3
4 \begin{document}
5   Si consideri la seguente \myfigure{figura}:
6   \includegraphics[width=3cm]{math-plot}
7   \captionof{figure}{Grafico costi/benefici}
```

Come si può notare, non contiene `\end{document}`; poi carica una classe disponibile solo all'autore; inoltre c'è un errore di sintassi nella riga 2 (`\usepackge` al posto di `\usepackage`); non si sa quale sia la definizione del comando `\myfigure`; non viene caricato il pacchetto `caption`, essenziale per poter usare il comando `\captionof`; infine, viene caricata un'immagine disponibile solo all'autore (`math-plot`).

## 3 Strumenti utili

### 3.1 Testo fittizio

Per inserire nel documento del testo fittizio, è bene ricorrere, se possibile, a uno dei molti pacchetti disponibili allo scopo, tra i quali ricordo `lipsum`, `blindtext` e `kantlipsum`: il codice sorgente risulta così molto più compatto.

Domanda 1: «Perché il primo capoverso non ha il rientro delle prima riga?»

```
\documentclass{book}
\usepackage{lipsum}

\begin{document}

  \section*{Stato dell'arte}
  \lipsum[1]

  \lipsum[2]

\end{document}
```

### 3.2 Immagini

Se il problema riguarda un'immagine, si possono percorrere due strade: (1) ricorrere all'opzione `demo` del pacchetto `graphicx`; (2) ricorrere alle immagini sempre disponibili nella propria distribuzione  $\text{\LaTeX}$ .

#### 3.2.1 L'opzione `demo`

Un esempio del primo caso è il seguente. Il file `math-plot.jpg`, evidentemente, è disponibile solo all'utente. Tuttavia, l'opzione `demo` permette in ogni caso di compilare il codice.

Domanda 2: «Perché la didascalia non compare e ottengo un errore?»

```
\documentclass{article}
\usepackage[demo]{graphicx}

\begin{document}

\includegraphics[width=4cm]{math-plot}
\caption{Grafico costi/benefici}

\end{document}
```

### 3.2.2 Immagini sempre disponibili

Un esempio del secondo caso è quest'altro. Invece di caricare uno dei propri file, si preferisce inserire un file disponibile nella distribuzione, evitando quindi l'opzione `demo`. Le immagini disponibili sono tutte quelle presenti nella cartella:

`usr/local/texlive/2020/texmf-dist/tex/latex/mwe`

Tra queste: `example-image-a`, `example-image-b`, `example-image-c`.

Domanda 3: «Come si fa per ruotare quest'immagine di 90 gradi?»

```
\documentclass{article}
\usepackage{graphicx}

\begin{document}

\includegraphics{example-image-a}

\end{document}
```

## 3.3 La classe standalone

La classe minimale standalone è molto utile per creare EMC elementari, ma non solo. Prevede molte opzioni, tra le quali le più utili sono a mio avviso le seguenti due:

**tikz** permette di inserire codice TikZ;

**border**= $\langle$ lunghezza $\rangle$  aggiunge un margine pari a  $\langle$ lunghezza $\rangle$ . Ad es. `border=5mm`.

Domanda 4: «Come si fa per aggiungere dello spazio attorno all'immagine?»

```
\documentclass[tikz]{standalone}

\begin{document}

\tikz \draw[fill=red] (0,0) rectangle (1,1);

\end{document}
```

### 3.4 L'ambiente filecontents

Spesso il problema da risolvere richiede l'uso di file esterni (file .bib, archivi .csv, eccetera). Mediante l'ambiente filecontents (il pacchetto filecontents non va *mai* caricato) è possibile inserirli direttamente nel codice dell'EMC, rendendolo così completamente autosufficiente:

```
\begin{filecontents}[<options>]{<name>.<ext>}

    contenuto del file <name>.<ext>

\end{filecontents}

\documentclass{book}
...
```

L'ambiente può essere usato prima o dopo `\begin{document}` e prevede due opzioni:

**overwrite** risulta utilissima per modificare dinamicamente il file esterno, senza doverlo eliminare prima di ogni nuova compilazione;

**noheader** impedisce di scrivere alcune righe di intestazione, che potrebbero generare errori in alcune situazioni (vedi esempio sotto).

L'ambiente è particolarmente utile per risolvere i problemi legati alla bibliografia (sez. 4) e le questioni riguardanti i comandi `\input` o `\include` (sez. 5) o pacchetti particolari come `datatool` che richiedono l'interazioni con archivi esterni in formato .csv o simile (sez. 6).

## 4 Esempi speciali per biblatex

Come è noto, le bibliografie composte con biblatex richiedono un archivio esterno .bib. Per creare un EMC utile ci sono due strade: (1) ricorrere all'archivio presente in ogni distribuzione; (2) ricorrere al proprio archivio personale.

### 4.1 Usare l'archivio standard di biblatex

Non sempre è necessario adoperare il proprio archivio bibliografico. Anzi, se il problema prescinde da quest'ultimo, si può ridurre l'esempio richiamando l'archivio bibliografico distribuito con la T<sub>E</sub>X Live: `biblatex-examples.bib`. Ecco un EMC a proposito:

Domanda 5: «Come si fa a mostrare le sole iniziali dei nomi degli autori?»

```
\documentclass{book}
\usepackage[style=authoryear]{biblatex}
\addbibresource{biblatex-examples.bib}

\begin{document}

    \textcite{knuth:ct}
    \printbibliography

\end{document}
```

## 4.2 Usare il proprio archivio bibliografico

Se il problema richiede espressamente un archivio bibliografico personale, ci viene in aiuto l'ambiente `filecontents`. È conveniente nominare l'archivio come `\jobname.bib` e passare all'ambiente l'opzione `overwrite`, utile nel caso in cui fosse necessario modificare il file `.bib`. Ecco un esempio:

Domanda 6: «Come ottenere il cognome corretto 'Rossi Campano, Mario'?»

```
\begin{filecontents}[overwrite]{\jobname.bib}
  @book{rossi:2009,
    author    = {Mario Rossi Campano},
    title     = {Storia della tipografia},
    publisher = {Editori Capitolini},
    location  = {Roma},
    year      = {2009}}
\end{filecontents}

\documentclass{book}
\usepackage[style=authoryear]{biblatex}
\addbibresource{\jobname.bib}

\begin{document}

  \textcite{rossi:2009}
  \printbibliography

\end{document}
```

## 5 Esempi con file richiamati da `\input` o `\include`

A volte il problema riguarda l'uso di file esterni richiamati nel file principale attraverso i comandi `\input` o `\include`. Anche in questo caso torna utile l'ambiente `filecontents`.

Domanda 7: «Qual è la differenza tra `\input` e `\include`?»

```
\begin{filecontents}[overwrite]{cap1.tex}
  \chapter{Vide quando draco venturus est}
  \lipsum
\end{filecontents}

\begin{filecontents}[overwrite]{cap2.tex}
  \chapter{Prega che vene lo papa santo}
  \lipsum
\end{filecontents}

\documentclass{book}
\usepackage{lipsum}

\begin{document}
```

```

\tableofcontents

\include{cap1}
\input{cap2}

\end{document}

```

## 6 Esempi con file .csv

Nell'esempio seguente, l'ambiente `filecontents` genera il file `database.csv` direttamente nella cartella di lavoro. L'opzione `noheader` assicura che all'inizio del file non venga aggiunta alcuna riga di intestazione.

Domanda 8: «Come si fa a includere anche l'indirizzo email?»

```

\begin{filecontents}[noheader,overwrite]{database.csv}
num, email, address, other, name
1, email1, Address 1, Other 1, Isaak Bacharach
2, email2, Address 2, Other 2, Reinhold Baer
3, email3, Address 3, Other 3, Christian Baer
4, email4, Address 4, Other 4, Wolf Barth
\end{filecontents}

\documentclass{book}
\usepackage{datatool}
\DTLloaddb{database}{database.csv}
\newcommand{\getauthor}[2]{%
  \DTLforeach*[\DTLisseq{\num}]{#1}
  {database}%etichetta del database
  {\num=num,\address=address,\name=name}%assegnazione
  {\textbf{\csname#2\endcsname}}}

\begin{document}

The name of the forth entry is \getauthor{4}{name} \\
The address of the second entry is \getauthor{2}{address}

\end{document}

```