

# La classe `verifica`\*

Riccardo Dossena†

4 luglio 2025

## Abstract

This `LATEX` document class is intended for teachers who must write out the text of a test or an exam. It provides various environments and commands to produce the typical exercises contained in this kind of document. It is mainly intended for Italian high school teachers, as the style is probably more in line with Italian high school tests.

## Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Licenza</b>	<b>2</b>
<b>3</b>	<b>Come usare la classe <code>verifica</code></b>	<b>2</b>
3.1	Opzioni di classe . . . . .	3
<b>4</b>	<b>L'intestazione della <code>verifica</code></b>	<b>3</b>
4.1	Altri tipi di intestazione . . . . .	5
4.2	Il nominativo del candidato . . . . .	6
<b>5</b>	<b>Gli ambienti</b>	<b>7</b>
5.1	L'ambiente <code>esercizi</code> . . . . .	7
5.2	Sottoelenchi dell'ambiente <code>esercizi</code> . . . . .	8
5.3	Sottoelenchi puntati . . . . .	9
5.4	Test a risposta multipla . . . . .	10
5.5	Scrivere su più colonne . . . . .	12
<b>6</b>	<b>Strumenti</b>	<b>13</b>
6.1	Vero o falso . . . . .	13
6.2	Righe per scrivere le risposte . . . . .	16
6.3	Griglia a quadretti . . . . .	16
6.4	Completamenti . . . . .	17

---

\*Versione v2.0 del 2025/07/04.

†E-mail: [riccardo.dossena@gmail.com](mailto:riccardo.dossena@gmail.com)

<b>7</b>	<b>Punteggio</b>	<b>18</b>
7.1	Comandi per inserire il punteggio . . . . .	18
7.2	Varianti per il punteggio . . . . .	20
7.3	Un piccolo bug del punteggio (e soluzione) . . . . .	21
7.4	Il riquadro del punteggio totale . . . . .	21
<b>8</b>	<b>Miscellanea</b>	<b>22</b>
8.1	Le frazioni doppie . . . . .	22
8.2	La virgola intelligente . . . . .	23
8.3	Soprassegno per gli archi . . . . .	23
8.4	Unità di misura . . . . .	24
8.5	Definizioni abbreviate . . . . .	25
<b>9</b>	<b>Implementation</b>	<b>26</b>
9.1	Class options . . . . .	26
9.2	The document heading . . . . .	28
9.3	New commands . . . . .	30
9.4	Test environments . . . . .	32
9.5	True/False tools . . . . .	32
9.6	Dotted lines . . . . .	34
9.7	Grids of squares . . . . .	35
9.8	Scoring tools . . . . .	35
<b>10</b>	<b>Change History</b>	<b>37</b>
<b>11</b>	<b>Index</b>	<b>37</b>

## 1 Introduzione

Questa classe di documento è rivolta principalmente agli insegnanti delle scuole superiori italiane che devono comporre il testo di una verifica, di un esame o di un test. Essa mette a disposizione vari ambienti e comandi per produrre gli esercizi tipici contenuti in questo genere di documenti.

Il codice è stato abbozzato in occasione dell’ultima lezione del Corso  $\text{\LaTeX}$  tenuto al Liceo Novello di Codogno (LO) fra ottobre e novembre 2019, prima sotto forma di pacchetto e in seguito di classe. L’intento era di fornire ai partecipanti uno strumento concreto e utile per la realizzazione di uno fra i documenti più frequentemente utilizzati: il test, comunemente chiamato “verifica”. Le attente osservazioni dei miei ottimi colleghi mi hanno spinto in seguito a potenziare la classe, al fine di permettere una migliore personalizzazione e un più efficace utilizzo degli ambienti definiti.

Desidero ringraziare Tommaso Gordini per i preziosi consigli e suggerimenti.

## 2 Licenza

Il codice  $\text{\LaTeX}$  di questa classe è rilasciato sotto la  $\text{\LaTeX}$  Project Public License, v1.3c.

## 3 Come usare la classe `verifica`

Per usare la classe è sufficiente copiare il file `verifica.cls` nella stessa cartella dove risiede il file sorgente del documento che si vuole comporre, oppure installarlo secondo le procedure standard in una directory opportuna della propria distribuzione T<sub>E</sub>X<sup>1</sup>. Per iniziare a editare un documento si può digitare direttamente il file `verifica-template.tex`, che contiene alcuni comandi già impostati.

Per caricare la classe, il comando da dare nel preambolo è

---

```
\documentclass{verifica}
```

---

### 3.1 Opzioni di classe

Le opzioni di default per il corpo del testo e per il formato della pagina sono “11pt” e “a4paper”. È possibile tuttavia modificarle inserendo altre opzioni di classe, ad esempio:

---

```
\documentclass[12pt,a5paper]{verifica}
```

---

Le opzioni di classe disponibili, oltre a quelle per il formato pagina, sono elencate di seguito.

<code>10pt</code>	corpo del testo a 10 pt
<code>11pt</code>	corpo del testo a 11 pt (default)
<code>12pt</code>	corpo del testo a 12 pt
<code>14pt</code>	corpo del testo a 14 pt
<code>senzagrazie</code>	stile del testo senza grazie (font <code>cmbright</code> )
<code>textbullet</code>	stile punti elenco • (anziché ►)
<code>intestazione semplice</code>	intestazione stile “semplice”
<code>intestazione righe</code>	intestazione stile “con righe”

## 4 L'intestazione della verifica

Per creare l'intestazione della verifica occorre impostare i comandi che ne definiscono la tipologia, la disciplina, l'istituto, la data, la classe ed eventualmente il tempo a disposizione (tabella 1).

Per generare l'intestazione si deve inserire il comando `\intestazione`.

La classe `verifica` fornisce inoltre i comandi `\tsa` e `\tso`<sup>2</sup> che generano rispettivamente i simboli <sup>a</sup> e <sup>o</sup> per i numeri ordinali, seguiti da uno spazio fine. Essi sono utili per scrivere l'argomento di `\classe`.

Mostriamo un esempio di intestazione standard:

---

```
\begin{document}

\tipologia{verifica orale}
```

---

<sup>1</sup>Si vedano, nel caso, le T<sub>E</sub>X FAQ alla pagina <https://texfaq.org/FAQ-inst-wlcf>.

<sup>2</sup>Abbreviazioni di `\textsuperscript{a}` e `\textsuperscript{o}`.

Comando	Esempi/descrizione	Valore di default
<code>\tipologia</code>	verifica, prova per competenze, test, prova comune, verifica orale, ...	verifica
<code>\disciplina</code>	matematica, fisica, scienze, italiano, ...	matematica
<code>\istituto</code>	nome dell'Istituto	Liceo "G. Novello" – Codogno
<code>\data</code>	data della verifica	<code>\today</code> (data odierna)
<code>\classe</code>	classe	<i>non definito</i>
<code>\tempo</code>	tempo a disposizione	<i>non definito</i>

Tabella 1: I comandi per l'intestazione della verifica.

```

\disciplina{fisica}
\istituto{Liceo "F. Indovino" -- Roma}
\classe{3\tsa C}
\data{12 novembre 2019}

```

```

\intestazione

```

<b>VERIFICA ORALE DI FISICA</b> <b>Liceo "F. Indovino" – Roma</b> <b>12 novembre 2019</b>	Cognome e nome  Classe 3 <sup>a</sup> C
---	---

Se in aggiunta si inserisce il comando `\tempo`, viene stampato il tempo a disposizione per la prova. In assenza dei comandi di definizione, apparirà l'intestazione coi valori di default indicati nella tabella 1. Ad esempio:

```

\begin{document}

\classe{}
\tempo{60 minuti}

\intestazione

```

<b>VERIFICA DI MATEMATICA</b> <b>Liceo “G. Novello” – Codogno</b> <b>4 luglio 2025</b>	Cognome e nome  Classe
<i>Tempo a disposizione: 60 minuti</i>	

Il campo `\classe` rimasto vuoto può rivelarsi utile nel caso in cui la verifica sia proposta a più classi e l'informazione debba essere inserita dal candidato.

Se il comando `\classe` non viene inserito, il campo corrispondente rimane indefinito e la dicitura “Classe” non appare.

---

```
\begin{document}
```

```
\intestazione
```

---

<b>VERIFICA DI MATEMATICA</b> <b>Liceo “G. Novello” – Codogno</b> <b>4 luglio 2025</b>	Cognome e nome
--	----------------

#### 4.1 Altri tipi di intestazione

È possibile usare altri due stili di intestazione tramite i comandi `\intestazione semplice` e `\intestazione righe`. La prima produce un'intestazione “essenziale”:

---

```
\tipologia{prova per competenze}
\disciplina{matematica}
\istituto{Liceo ‘‘G. Novello’’ -- Codogno (LO)}
\classe{3\tsa C}
\data{12 novembre 2019}
```

```
\intestazione semplice
```

---

Liceo “G. Novello” – Codogno	Classe 3 <sup>a</sup> C	12 novembre 2019
<b>PROVA PER COMPETENZE DI MATEMATICA</b>		

La seconda produce un'intestazione delimitata da due righe:

---

```
\tipologia{verifica orale}
```

```

\disciplina{matematica}
\istituto{Liceo ‘‘G. Novello’’ -- Codogno (LO)}
\classe{2\tsa B}
\data{12 novembre 2019}
\tempo{55 minuti}

```

```
\intestazionerighe
```

Liceo ‘‘G. Novello’’ – Codogno	Classe 2 <sup>a</sup> B	12 novembre 2019
<b>VERIFICA ORALE DI MATEMATICA</b>		
<i>Tempo a disposizione: 55 minuti</i>		

In entrambi i casi si può definire oppure no il tempo a disposizione.

Anziché usare i comandi `\intestazione semplice` e `\intestazionerighe`, è possibile impostare un’opzione di classe. Scrivendo infatti

```
\documentclass[intestazione semplice]{verifica}
```

oppure

```
\documentclass[intestazionerighe]{verifica}
```

l’intestazione verrà generata dal comando `\intestazione` con lo stile impostato dall’opzione.

## 4.2 Il nominativo del candidato

Le intestazioni ‘‘semplice’’ e ‘‘con righe’’ non prevedono spazi in cui il candidato deve scrivere il proprio nome. Il comando `\lineanome` genera una linea punteggiata con la dicitura ‘‘Cognome e nome’’ destinata a tale scopo.

```

\tipologia{verifica orale}
\disciplina{matematica}
\istituto{Liceo ‘‘G. Novello’’ -- Codogno (LO)}
\classe{2\tsa B}
\data{12 novembre 2019}
\tempo{55 minuti}

\lineanome
\intestazione semplice

```

Cognome e nome .....		
Liceo “G. Novello” – Codogno	Classe 2 <sup>a</sup> B	12 novembre 2019
<b>VERIFICA ORALE DI MATEMATICA</b>		
<i>Tempo a disposizione: 55 minuti</i>		

Di seguito il risultato per l'intestazione “con righe”.

Cognome e nome .....		
Liceo “G. Novello” – Codogno	Classe 2 <sup>a</sup> B	12 novembre 2019
<b>VERIFICA ORALE DI MATEMATICA</b>		
<i>Tempo a disposizione: 55 minuti</i>		

Se l'intestazione è quella di default, che prevede di per sé lo spazio per il nome del candidato, il comando `\lineanome` produce semplicemente una riga punteggiata nel riquadro di destra.

<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;"><b>VERIFICA ORALE DI MATEMATICA</b></td> </tr> <tr> <td style="text-align: center;">Liceo “G. Novello” – Codogno</td> </tr> <tr> <td style="text-align: center;">12 novembre 2019</td> </tr> </table>	<b>VERIFICA ORALE DI MATEMATICA</b>	Liceo “G. Novello” – Codogno	12 novembre 2019	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>Cognome e nome</td> </tr> <tr> <td>.....</td> </tr> <tr> <td>Classe 2<sup>a</sup> B</td> </tr> </table>	Cognome e nome	.....	Classe 2 <sup>a</sup> B
<b>VERIFICA ORALE DI MATEMATICA</b>							
Liceo “G. Novello” – Codogno							
12 novembre 2019							
Cognome e nome							
.....							
Classe 2 <sup>a</sup> B							
<i>Tempo a disposizione: 55 minuti</i>							

## 5 Gli ambienti

### 5.1 L'ambiente esercizi

L'ambiente `esercizi` è quello principale: esso produce, mediante il pacchetto `enumitem`, un elenco numerato allineato al margine sinistro, con etichette (label) in grassetto. La spaziatura verticale fra un esercizio (item) e l'altro è impostata per default a 2 em.

---

```

\begin{esercizi}
\item La mamma dà a Pierino \EUR{10} per comprare la frutta. Pierino li
    spende tutti. Quanto rimane a Pierino?

\item Risolvi la seguente equazione
    \[
    x+1=2
    \]

\item Enuncia e dimostra il teorema di Pitagora.
\end{esercizi}

```

---

1. La mamma dà a Pierino 10 € per comprare la frutta. Pierino li spende tutti. Quanto rimane a Pierino?
2. Risolvi la seguente equazione
$$x + 1 = 2$$
3. Enuncia e dimostra il teorema di Pitagora.

Qualora si volesse modificare la spaziatura tra gli item, è sufficiente modificare il valore del parametro `itemsep`, esattamente come nelle liste definite da `enumitem`. Nel codice di esempio seguente viene prodotta una spaziatura di 1 em (anziché di 2 em) tra un item e l'altro.

---

```
\begin{esercizi}[itemsep=1em]
\item La mamma dà a Pierino...
...
\end{esercizi}
```

---

## 5.2 Sottoelenchi dell'ambiente `esercizi`

È possibile, all'interno dell'ambiente `esercizi`, creare dei sottoelenchi che sfruttano i vantaggi del pacchetto `enumitem`. Mostriamo come esempio un esercizio che consiste in un elenco di equazioni da risolvere.

---

```
\begin{esercizi}
\item Risolvi le seguenti equazioni
  \begin{enumerate}[a],itemsep=1.5em,topsep=1.2em,leftmargin=*
```

$$\frac{x+1}{x-2} + 4 = \frac{x}{x-1}$$

```
  \item  $\sin^2 x + \cos x = \cos 2x$ 
  \item  $\ln(x+1) = \ln 5 + \log_2 x$ 
\end{enumerate}
\end{esercizi}
```

---

1. Risolvi le seguenti equazioni

a)  $\frac{x+1}{x-2} + 4 = \frac{x}{x-1}$

b)  $\sin^2 x + \cos x = \cos 2x$

c)  $\ln(x+1) = \ln 5 + \log_2 x$

Il comando `\dss` è un'abbreviazione di `\displaystyle`, che serve a ottenere le formule nel testo in stile display. La tabella 2 mostra alcuni dei parametri opzionali di `enumitem` che possono essere impostati per personalizzare gli elenchi.

Parametro	Esempi/descrizione
a)	imposta la numerazione alfabetica con parentesi. Altri esempi sono: <ul style="list-style-type: none"> <li>• A. (numerazione alfabetica in maiuscolo con punto)</li> <li>• i. (numerazione romana in minuscolo con punto)</li> <li>• <code>\bfseries 1)</code> (numerazione in grassetto con parentesi)</li> </ul>
<code>itemsep</code>	imposta lo spazio verticale tra un item e l'altro
<code>topsep</code>	imposta lo spazio verticale tra il testo che precede l'elenco e il primo item
<code>labelsep</code>	imposta la distanza tra il testo dell'item e l'etichetta
<code>leftmargin</code>	imposta il rientro dell'elenco rispetto al margine sinistro; il comando <code>leftmargin=*</code> allinea le etichette (label) numeriche al margine degli item dell'elenco principale

Tabella 2: Alcuni parametri utili del pacchetto `enumitem`.

### 5.3 Sottoelenchi puntati

È possibile usare `itemize` per suddividere un esercizio in punti non numerati, utilizzando gli stessi parametri opzionali dell'ambiente `enumerate`.

```
\begin{esercizi}
\item Un'automobile sta viaggiando alla velocità  $v_0$  lungo una strada
rettilenea. A un certo istante  $t_0$  comincia ad accelerare e
raggiunge la velocità  $v$  all'istante  $t$ . Calcola:
\begin{itemize}
\item l'accelerazione media dell'automobile;
\item lo spazio percorso in fase di accelerazione.
\end{itemize}
\end{esercizi}
```

1. Un'automobile sta viaggiando alla velocità  $v_0$  lungo una strada rettilinea. A un certo istante  $t_0$  comincia ad accelerare e raggiunge la velocità  $v$  all'istante  $t$ . Calcola:
  - ▶ l'accelerazione media dell'automobile;
  - ▶ lo spazio percorso in fase di accelerazione.

Se al posto di ▶ si preferisce •, si deve impostare l'opzione di classe `itembullet`. Nell'esempio seguente mostriamo anche l'uso del parametro `leftmargin=*`, che evita il rientro del sottoelenco.

---

```

\documentclass[itmbullet]{verifica}
...
\begin{esercizi}
\item Un'automobile sta viaggiando...
...
\begin{itemize}[leftmargin=*]
\item l'accelerazione media dell'automobile;
\item lo spazio percorso in fase di accelerazione.
\end{itemize}
\end{esercizi}

```

---

1. Un'automobile sta viaggiando alla velocità  $v_0$  lungo una strada rettilinea. A un certo istante  $t_0$  comincia ad accelerare e raggiunge la velocità  $v$  all'istante  $t$ . Calcola:
  - l'accelerazione media dell'automobile;
  - lo spazio percorso in fase di accelerazione.

## 5.4 Test a risposta multipla

Vengono forniti tre ambienti per i test a risposta multipla: l'ambiente `test` (risposte allineate verticalmente) e gli ambienti “in linea” `test-orizz` e `test-orizz-newline` (“orizzontali”, cioè con risposte scritte di seguito). Mostriamo di seguito come utilizzarli.

---

```

\begin{esercizi}
\item Quale tra le funzioni seguenti è la derivata di  $1 + \sin x$ ?
\begin{test}
\item  $\sin x$ ;
\item  $\cos x$ ;
\item  $-\sin x$ ;
\item  $-\cos x$ ;
\item nessuna delle precedenti.
\end{test}
\end{esercizi}

```

---

1. Quale tra le funzioni seguenti è la derivata di  $1 + \sin x$ ?

- $a$   $\sin x$ ;
- $b$   $\cos x$ ;
- $c$   $-\sin x$ ;
- $d$   $-\cos x$ ;
- $e$  nessuna delle precedenti.

---

```

\begin{esercizi}

```

```

\item Quale tra le funzioni seguenti è la derivata di  $1+\sin x$ ?
  \begin{test-orizz}
    \item  $\sin x$ ;
    \item  $\cos x$ ;
    \item  $-\sin x$ ;
    \item  $-\cos x$ ;
    \item nessuna delle precedenti.
  \end{test-orizz}
\end{esercizi}

```

1. Quale tra le funzioni seguenti è la derivata di  $1 + \sin x$ ?   $a$   $\sin x$ ;   $b$   $\cos x$ ;  
  $c$   $-\sin x$ ;   $d$   $-\cos x$ ;   $e$  nessuna delle precedenti.

Se si desidera che le risposte comincino a capo in una nuova riga, è disponibile la variante `test-orizz-newline`:

```

\begin{esercizi}
\item Quale tra le funzioni seguenti è la derivata di  $1+\sin x$ ?
  \begin{test-orizz-newline}
    \item  $\sin x$ ;
    \item  $\cos x$ ;
    \item  $-\sin x$ ;
    \item  $-\cos x$ ;
    \item nessuna delle precedenti.
  \end{test-orizz-newline}
\end{esercizi}

```

1. Quale tra le funzioni seguenti è la derivata di  $1 + \sin x$ ?  
  $a$   $\sin x$ ;   $b$   $\cos x$ ;   $c$   $-\sin x$ ;   $d$   $-\cos x$ ;   $e$  nessuna delle precedenti.

Gli ambienti `test`, `test-orizz` e `test-orizz-newline` sono ancora liste definite tramite il pacchetto `enumitem`, per cui è possibile modificarne i parametri come mostrato per l'ambiente `esercizi`<sup>3</sup>.

Se si vuole cambiare il formato delle label per l'intero documento, si può impostare nel preambolo o all'inizio del testo il comando seguente:

```
\labeltest{\label}
```

dove `\label` definisce il nuovo stile secondo i criteri di `enumitem`, anche in modalità `shortlabels`. Ricordiamo che se il contatore viene scritto tra parentesi graffe (ad esempio se si utilizza una box) non può essere in forma abbreviata, ma deve essere una fra `\arabic*`, `\alph*`, `\Alph*`, `\roman*` o `\Roman*`.

L'esempio che segue è ispirato ai testi dei *Giocchi di Archimede*.

<sup>3</sup>Consultare, nel caso, la documentazione di `enumitem`: <http://ctan.mirror.garr.it/mirrors/CTAN/macros/latex/contrib/enumitem/enumitem.pdf>.

---

```

\labeltest{\bfseries (A)}
...
\begin{esercizi}
\item Quale tra le funzioni seguenti è la derivata di  $1+\sin x$ ?
  \begin{test-orizz-newline}
    \item  $\sin x$ ;
    \item  $\cos x$ ;
    \item  $-\sin x$ ;
    \item  $-\cos x$ ;
    \item nessuna delle precedenti.
  \end{test-orizz-newline}
\end{esercizi}

```

---

1. Quale tra le funzioni seguenti è la derivata di  $1 + \sin x$ ?
- (A)  $\sin x$ ; (B)  $\cos x$ ; (C)  $-\sin x$ ; (D)  $-\cos x$ ; (E) nessuna delle precedenti.

La classe verifica definisce i comandi `\labelbox` e `\biglabelbox`, che racchiudono i loro argomenti in box quadrate “framed”. La prima è usata per le label di default dei test a risposta multipla e la seconda per le caselle dei test vero/falso. Esse possono essere opportunamente utilizzate insieme a `\labeltest` per ridefinire lo stile di altre label.

---

```

\labeltest{\biglabelbox{\Alph*}}
...
\begin{esercizi}
\item Quale tra le funzioni seguenti è la derivata di  $1+\sin x$ ?
  \begin{test-orizz-newline}
    \item  $\sin x$ ;
    \item  $\cos x$ ;
    \item  $-\sin x$ ;
    \item  $-\cos x$ ;
    \item nessuna delle precedenti.
  \end{test-orizz-newline}
\end{esercizi}

```

---

1. Quale tra le funzioni seguenti è la derivata di  $1 + \sin x$ ?
- A  $\sin x$ ;  B  $\cos x$ ;  C  $-\sin x$ ;  D  $-\cos x$ ;  E nessuna delle precedenti.

## 5.5 Scrivere su più colonne

La classe carica in automatico il pacchetto `multicol`, che consente di scrivere su più colonne tramite l’ambiente `multicols`:

---

```

\begin{multicols}{(numero di colonne)}
...
\end{multicols}

```

---

L'esempio seguente mostra un possibile utilizzo di questa funzionalità.

```
{\setlength{\columnsep}{2.5em}
\begin{multicols}{2}
\begin{esercizi}
  \item Quale tra le funzioni seguenti è la derivata di  $1+\sin x$ ?
    \begin{test}
      \item  $\sin x$ ;
      \item  $\cos x$ ;
      \item  $-\sin x$ ;
      \item  $-\cos x$ ;
      \item nessuna delle precedenti.
    \end{test}\columnbreak
  \item Un'automobile sta viaggiando...
  ...
  \begin{itemize}[leftmargin=*]
    \item l'accelerazione media dell'automobile;
    \item lo spazio percorso in fase di accelerazione.
  \end{itemize}
\end{esercizi}
\end{multicols}}
```

<p>1. Quale tra le funzioni seguenti è la derivata di <math>1 + \sin x</math>?</p> <p><input type="checkbox"/> <i>a</i> <math>\sin x</math>;</p> <p><input type="checkbox"/> <i>b</i> <math>\cos x</math>;</p> <p><input type="checkbox"/> <i>c</i> <math>-\sin x</math>;</p> <p><input type="checkbox"/> <i>d</i> <math>-\cos x</math>;</p> <p><input type="checkbox"/> <i>e</i> nessuna delle precedenti.</p>	<p>2. Un'automobile sta viaggiando alla velocità <math>v_0</math> lungo una strada rettilinea. A un certo istante <math>t_0</math> comincia ad accelerare e raggiunge la velocità <math>v</math> all'istante <math>t</math>. Calcola:</p> <p>► l'accelerazione media dell'automobile;</p> <p>► lo spazio percorso in fase di accelerazione.</p>
---	---

L'esempio mostra anche come usare il comando `\setlength{\columnsep}{\langle valore \rangle}` per impostare lo spazio di separazione tra le colonne. Questo comando e tutto l'ambiente `multicols` vanno racchiusi tra parentesi graffe, in modo che lo spazio di separazione impostato abbia effetto solo sull'ambiente multicolonna corrente. Se lo si desidera, si può far apparire una linea di separazione tra le colonne con `\setlength{\columnseprule}{.4pt}`.

## 6 Strumenti

### 6.1 Vero o falso

Se la lingua selezionata per il documento è l'italiano o un'altra lingua diversa dall'inglese, il comando `\vf` stampa alla fine della riga corrente le box  **V**  **F**. Se invece la lingua selezionata è l'inglese (per l'intero documento o solo per una parte, dopo `\selectlanguage{english}` o all'interno dell'ambiente `otherlanguage`) stampa  **T**  **F**.

Questo comando è utile per test con brevi domande vero/falso.

---

```

\begin{esercizi}
\item Stabilisci se le seguenti affermazioni sono vere o false.
\begin{enumerate}[a]
\item La derivata di  $3x^2$  è  $6x$ . \vf
\item  $7+3=11$ . \vf
\item Una soluzione di  $x^2-1=0$  è  $1$ . \vf
\item Il lavoro di una forza costante  $\vec{F}$  relativo allo spostamento
 $\vec{s}$  è dato dal prodotto scalare  $\vec{F}\cdot\vec{s}$ . \vf
\end{enumerate}
\end{esercizi}

```

---

1. Stabilisci se le seguenti affermazioni sono vere o false.

a) La derivata di $3x^2$ è $6x$ .	<input type="checkbox"/> V <input type="checkbox"/> F
b) $7 + 3 = 11$ .	<input type="checkbox"/> V <input type="checkbox"/> F
c) Una soluzione di $x^2 - 1 = 0$ è $1$ .	<input type="checkbox"/> V <input type="checkbox"/> F
d) Il lavoro di una forza costante $\vec{F}$ relativo allo spostamento $\vec{s}$ è dato dal prodotto scalare $\vec{F} \cdot \vec{s}$ .	<input type="checkbox"/> V <input type="checkbox"/> F

Come si vede, se la frase è troppo lunga e raggiunge il margine destro, il risultato è insoddisfacente. Per rimediare all'inconveniente viene definito l'ambiente `test-verofalso`. Esso produce una lista con label alfabetiche e item che si interrompono al 75% della lunghezza della linea. Viene inoltre stampato automaticamente  V  F alla fine dell'ultima riga di ogni item. Ciascun item deve essere impostato tramite il comando `\vfitem{<testo>}`<sup>4</sup>.

---

```

\begin{esercizi}
\item Stabilisci se le seguenti affermazioni sono vere o false.
\begin{test-verofalso}
\vfitem{La derivata di  $3x^2$  è  $6x$ .}
...
\vfitem{Il lavoro di una forza costante  $\vec{F}$  relativo allo
spostamento  $\vec{s}$  è dato dal prodotto scalare
 $\vec{F}\cdot\vec{s}$ .}
\vfitem{Siano  $f$  e  $g$  due funzioni reali di variabile reale
aventi lo stesso dominio. Se esse sono entrambe pari oppure entrambe
dispari allora  $fg$  è una funzione pari; se una di esse è dispari
allora  $fg$  è una funzione dispari.}
\end{test-verofalso}
\end{esercizi}

```

---

<sup>4</sup>La versione 1.0 della classe prevedeva la creazione dei test vero/falso mediante il comando `\verofalso`. Questo comando è ancora attivo, per cui rimane piena compatibilità con la versione 1.0.

1. Stabilisci se le seguenti affermazioni sono vere o false.

a) La derivata di  $3x^2$  è  $6x$ .  V  F

b)  $7 + 3 = 11$ .  V  F

c) Una soluzione di  $x^2 - 1 = 0$  è 1.  V  F

d) Il lavoro di una forza costante  $\vec{F}$  relativo allo spostamento  $\vec{s}$  è dato dal prodotto scalare  $\vec{F} \cdot \vec{s}$ .  V  F

e) Siano  $f$  e  $g$  due funzioni reali di variabile reale aventi lo stesso dominio. Se esse sono entrambe pari oppure entrambe dispari allora  $fg$  è una funzione pari; se una di esse è dispari allora  $fg$  è una funzione dispari.  V  F

L'ambiente `test-verofalso` è ancora una lista creata tramite `enumitem`, per cui è possibile impostare per essa tutti i parametri descritti nella tabella 2. Ad esempio:

---

```
\begin{test-verofalso}[1],itemsep=1.2em,leftmargin=*,...]
```

---

Se anziché al 75% si vuole interrompere la riga a una percentuale diversa, si può scegliere il fattore di scala desiderato (riferito a 1) impostando il parametro `fattorevf`. Nell'esempio che segue impostiamo l'interruzione all'85% della lunghezza della riga.

---

```
\begin{esercizio}
\item Stabilisci se le seguenti affermazioni sono vere o false.
\begin{test-verofalso}[fattorevf=.85]
\vfitem{Il lavoro di una forza costante  $\vec{F}$  relativo
allo spostamento  $\vec{s}$  è dato dal prodotto scalare
 $\vec{F} \cdot \vec{s}$ .}
\vfitem{Siano  $f$  e  $g$  due funzioni reali di variabile reale
aventi lo stesso dominio. Se esse sono entrambe pari oppure
entrambe dispari allora  $fg$  è una funzione pari; se una di
esse è dispari allora  $fg$  è una funzione dispari.}
\end{test-verofalso}
\end{esercizio}
```

---

1. Stabilisci se le seguenti affermazioni sono vere o false.

a) Il lavoro di una forza costante  $\vec{F}$  relativo allo spostamento  $\vec{s}$  è dato dal prodotto scalare  $\vec{F} \cdot \vec{s}$ .  V  F

b) Siano  $f$  e  $g$  due funzioni reali di variabile reale aventi lo stesso dominio. Se esse sono entrambe pari oppure entrambe dispari allora  $fg$  è una funzione pari; se una di esse è dispari allora  $fg$  è una funzione dispari.  V  F

Infine, se si vuole impostare definitivamente un fattore di scala per l'intero documento, senza doverlo specificare ogni volta come parametro, si può inserire nel preambolo o all'inizio del testo il comando:

---

`\fattorevf{<fattore di scala>}`

---

dove *<fattore di scala>* deve essere un numero compreso tra 0 e 1.

## 6.2 Righe per scrivere le risposte

Viene definito il comando:

---

`\riga{<numero di righe>}`

---

che stampa delle righe punteggiate ben spaziate verticalmente, su cui il candidato può scrivere manualmente (tante righe quante indicate nell'argomento).

---

```
\begin{esercizi}
  \item Enuncia il teorema di Pitagora.\par\riga{3}
\end{esercizi}
```

---

<p>1. Enuncia il teorema di Pitagora.</p> <p>.....</p> <p>.....</p> <p>.....</p>
--

Se si omette `\par` (che fa andare a capo e iniziare un nuovo capoverso) e si inserisce il comando di seguito al testo, la prima riga viene stampata dalla posizione corrente fino al margine destro.

---

```
\begin{esercizi}
  \item Enuncia il teorema di Pitagora. \riga{3}
\end{esercizi}
```

---

<p>1. Enuncia il teorema di Pitagora. ....</p> <p>.....</p> <p>.....</p>
--

## 6.3 Griglia a quadretti

Se si desidera inserire una griglia a quadretti, è possibile usare il comando:

---

`\drawgrid[<dimensione quadretto>]{<altezza griglia>}`

---

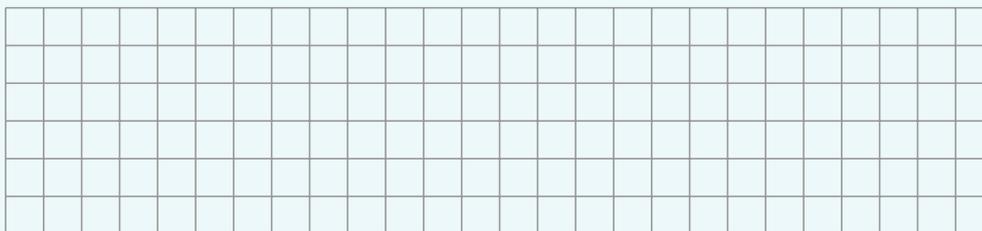
Questo comando disegna una griglia a quadretti con altezza specificata e larghezza pari alla lunghezza della linea corrente. La dimensione predefinita dei quadretti è 0,5 cm, ma può essere modificata tramite l'argomento opzionale. Se necessario, l'altezza della griglia viene arrotondata per difetto al multiplo più vicino della dimensione del quadretto che non superi il valore specificato.

---

```
\begin{esercizi}
  \item Trova la velocità media di un'auto che percorre  $30\text{ km}$  in
     $60\text{ min}$ . \drawgrid{3cm}
\end{esercizi}
```

---

1. Trova la velocità media di un'auto che percorre 30 km in 60 min.



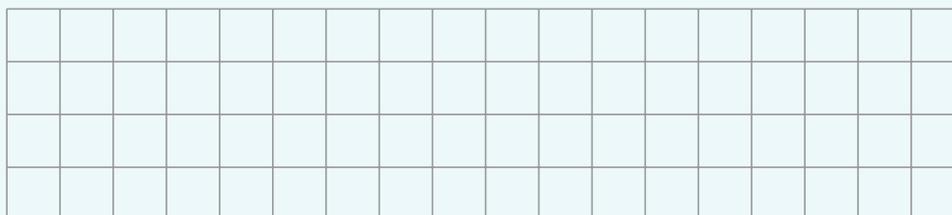
Per ottenere quadretti di dimensione diversa, è sufficiente specificarla nell'argomento opzionale:

---

```
\begin{esercizi}
  \item Trova la velocità media di un'auto che percorre  $30\text{ km}$  in
     $60\text{ min}$ . \drawgrid[0.7cm]{3cm}
\end{esercizi}
```

---

1. Trova la velocità media di un'auto che percorre 30 km in 60 min.



Questo comando disegna una griglia della stessa altezza ma con quadretti di lato 0,7 cm.

## 6.4 Completamenti

Se un esercizio consiste nel completamento di un testo con alcune parole chiave, è possibile usare i comandi `\dotword` e `\dotrule`. Il primo produce una linea punteggiata di lunghezza doppia della parola racchiusa come argomento (cioè la parola esatta che lo studente dovrebbe inserire), mentre il secondo produce una linea punteggiata di lunghezza data nell'argomento.

---

```
\begin{esercizi}
```

```

\item \emph{Completa il seguente enunciato.} La potenza di
\dotword{potenza} è una potenza che ha per \dotword{base}
la stessa base e per \dotword{esponente} il \dotword{prodotto}
degli esponenti.
\end{esercizi}

```

1. *Completa il seguente enunciato.* La potenza di ..... è una potenza che ha per ..... la stessa base e per ..... il ..... degli esponenti.

Notiamo che in questo modo il file sorgente rende possibile la lettura esplicita delle parole esatte, con un notevole vantaggio per la chiarezza.

Se è necessario aumentare lo spazio tra una riga e l'altra, si può usare `\doublespacing` (dal pacchetto `setspace`, caricato in automatico). Per ripristinare l'impostazione iniziale, basta inserire `\par\singlespacing`, come nell'esempio seguente.

```

\begin{esercizi}
\item \doublespacing\emph{Completa il seguente enunciato.} La potenza di
\dotword{potenza} è una potenza che ha per \dotword{base} la
stessa base e per \dotword{esponente} il \dotword{prodotto} degli
esponenti. \par\singlespacing
\end{esercizi}

```

1. *Completa il seguente enunciato.* La potenza di ..... è una potenza che ha per ..... la stessa base e per ..... il ..... degli esponenti.

Riproponiamo lo stesso esempio usando il comando `\dotrule{4cm}`, che produce linee punteggiate lunghe 4 cm.

```

\begin{esercizi}
\item \doublespacing\emph{Completa il seguente enunciato.} La potenza di
\dotrule{4cm} è una potenza che ha per \dotrule{4cm} la stessa base
e per \dotrule{4cm} il \dotrule{4cm} degli esponenti.
\par\singlespacing
\end{esercizi}

```

1. *Completa il seguente enunciato.* La potenza di ..... è una potenza che ha per ..... la stessa base e per ..... il ..... degli esponenti.

## 7 Punteggio

### 7.1 Comandi per inserire il punteggio

Se si desidera assegnare un punteggio ad ogni esercizio, è possibile utilizzare il comando:

---

```
\punti[<testo aggiuntivo>]{<punteggio>}
```

---

Esso stampa il punteggio relativo all'esercizio, allineandolo al margine destro sulla riga corrente.

---

```
\begin{esercizi}
\item Data la retta  $y=2x$ , scrivi l'equazione di una sua
perpendicolare. \punti{10}
\item Data la retta  $y=3x-1$ , calcola l'equazione della retta per  $A$  ad essa
parallela. \punti{10}
\item Data la circonferenza di equazione  $x^2+y^2-3x+y-2=0$ , determina
le coordinate del centro, il suo raggio e calcola la
lunghezza della circonferenza. \punti{20}
\end{esercizi}
```

---

- |  |
|--|
| <p>1. Data la retta <math>y = 2x + 1</math>, scrivi l'equazione di una sua perpendicolare. [PUNTI 10]</p> <p>2. Data la retta <math>y = 3x - 1</math>, calcola l'equazione della retta per <math>A</math> ad essa parallela. [PUNTI 10]</p> <p>3. Data la circonferenza di equazione <math>x^2 + y^2 - 3x + y - 2 = 0</math>, determina le coordinate del centro, il suo raggio e calcola la lunghezza della circonferenza. [PUNTI 20]</p> |
|--|

Se lo spazio disponibile sulla riga non fosse sufficiente, il punteggio viene scritto automaticamente nella riga successiva.

L'opzione *<testo aggiuntivo>* serve per specificare eventuali informazioni da stampare dopo il punteggio. Ciò può essere utile ad esempio per indicare uno stesso punteggio per più esercizi.

---

```
\begin{esercizi}
\item Risolvi le seguenti equazioni. \punti[per ogni equazione]{10}
\begin{enumerate}[a],itemsep=.5em
\item  $x^2-3x=1$ 
\item  $2x+1=4x^2-3$ 
\end{enumerate}
\end{esercizi}
```

---

- |   |
|---|
| <p>1. Risolvi le seguenti equazioni. [PUNTI 10 PER OGNI EQUAZIONE]</p> <p>a) <math>x^2 - 3x = 1</math></p> <p>b) <math>2x + 1 = 4x^2 - 3</math></p> |
|---|

Nel caso particolare in cui si volesse stampare il punteggio alla stessa altezza di una formula in display, è possibile inserirlo come argomento del comando `\tag*`.

---

```
\begin{esercizi}
```

```
\item Disegna la parabola di equazione
\begin{equation}
x=-\frac{1}{2}x^2-x+1 \tag*{\punti{5}}
\end{equation}
\end{esercizi}
```

1. Disegna la parabola di equazione

$$y = -\frac{1}{2}x^2 - x + 1$$

[PUNTI 5]

## 7.2 Varianti per il punteggio

Esistono due varianti per la stampa del punteggio, utilizzabili anche simultaneamente.

La prima si ottiene mediante il comando-dichiarazione `\puntiadestra`, che imposta la stampa del valore numerico del punteggio prima della dicitura “PUNTI”.

```
\puntiadestra
\begin{esercizi}
\item Scrivi l'equazione di una retta parallela all'asse $y$. \punti{5}
\end{esercizi}
```

1. Scrivi l'equazione di una retta parallela all'asse  $y$ .

[5 PUNTI]

La seconda si ottiene col comando-dichiarazione `\punti`, che sostituisce alla scritta “PUNTI” l'abbreviazione “PT.”

```
\punti
\begin{esercizi}
\item Scrivi l'equazione di una retta parallela all'asse $y$. \punti{5}
\end{esercizi}
```

1. Scrivi l'equazione di una retta parallela all'asse  $y$ .

[PT. 5]

Combinando le due dichiarazioni si ha:

```
\puntiadestra\punti
\begin{esercizi}
\item Scrivi le equazioni di due rette parallele
all'asse $y$. \punti[per ogni retta]{5}
\end{esercizi}
```

1. Scrivi le equazioni di due rette parallele all'asse  $y$ .

[5 PT. PER OGNI RETTA]

### 7.3 Un piccolo bug del punteggio (e soluzione)

Se la prima parola di una riga dove si intende stampare il punteggio è sillabata, il comando `\punti` sposta il punteggio nella riga vuota successiva, quindi in una posizione non corretta, come mostra il prossimo esempio.

```
\begin{esercizi}
  \item Scrivi in forma implicita l'equazione della retta parallela
    all'asse~ $x$  e precipitevolissimamente passante
    per  $P(0, 1)$ . \punti{12}
\end{esercizi}
```

1. Scrivi in forma implicita l'equazione della retta parallela all'asse  $x$  e precipitevolissimamente passante per  $P(0, 1)$ .

[PUNTI 12]

Qualora si presentasse una situazione del genere, è possibile risolvere sostituendo localmente al comando `\punti` il comando `\puntiman` (che sta per “punteggio manuale” e ha la stessa sintassi di `\punti`).

```
\begin{esercizi}
  \item Scrivi in forma implicita l'equazione della retta parallela
    all'asse~ $x$  e precipitevolissimamente passante
    per  $P(0, 1)$ . \puntiman{12}
\end{esercizi}
```

1. Scrivi in forma implicita l'equazione della retta parallela all'asse  $x$  e precipitevolissimamente passante per  $P(0, 1)$ .

[PUNTI 12]

### 7.4 Il riquadro del punteggio totale

Se lo si desidera, è possibile stampare un riquadro in cui inserire il punteggio totale della verifica, specificando il calcolo del voto finale. Il comando, da dare al di fuori dell'ambiente `esercizi`, è

```
\totpunti
```

Esso produce, in fondo alla pagina e allineato al margine destro, il riquadro seguente:

<b>Totale punti</b> <input type="text"/> /10 + 1 = <input type="text"/>	<b>VOTO</b>
---	-------------

L'impostazione di default prevede che il punteggio massimo totalizzabile sia 90 punti e che il voto (in decimi) sia calcolato tramite la formula

$$\text{Voto} = \text{Punteggio totalizzato}/10 + 1$$

Per impostare un calcolo diverso, basta inserirlo come argomento opzionale di `\totpunti`.

---

```
\begin{document}
\begin{esercizi}
...
\end{esercizi}
\totpunti[/12+2]
\end{document}
```

---

**VOTO**

Totale punti  /12 + 2 =

La variante `\vartotpunti` produce lo stesso riquadro senza spostarlo in fondo alla pagina. In questo modo risulta possibile posizionare un altro oggetto appena prima del riquadro stesso (ad esempio una tabella dei punteggi ottenuti), inserendo manualmente eventuali comandi `\vfill`.

## 8 Miscellanea

### 8.1 Le frazioni doppie

Le frazioni “doppie” (frazioni di frazioni) non vengono rese in modo ottimale con i comandi standard di  $\text{\LaTeX}$ . Si noti come il codice

---

```
\[
\frac{\dfrac{3}{2}+\dfrac{4}{3}}{\dfrac{5}{7}+\dfrac{8}{17}+1}
\]
```

---

produca frazioni non ben distanziate dalla linea di frazione principale:

$$\frac{\frac{3}{2} + \frac{4}{3}}{\frac{5}{7} + \frac{8}{17} + 1}$$

Per rimediare, la classe `verifica` mette a disposizione i comandi `\ddfrac`, `\numfrac` e `\denfrac`. Il primo imposta la frazione principale, con una linea leggermente sporgente; il secondo e il terzo servono per inserire frazioni al numeratore e al denominatore distanziate dalla linea di frazione principale. In presenza di più frazioni al numeratore o al denominatore, è sufficiente usare questi comandi solo per la prima e inserire le successive con `\frac`.

---

```
\[
\ddfrac{\numfrac{3}{2}+\frac{4}{3}}{\denfrac{5}{7}+\frac{8}{17}+1} \quad \quad
\ddfrac{\numfrac{3}{2}+2+\frac{1}{3}}{4} \quad \quad
\ddfrac{4}{\denfrac{1}{2}+\frac{2}{7}}
\]
```

---

$$\frac{\frac{3}{2} + \frac{4}{3}}{\frac{5}{7} + \frac{8}{17} + 1} \quad \frac{\frac{3}{2} + 2 + \frac{1}{3}}{4} \quad \frac{4}{\frac{1}{2} + \frac{2}{7}}$$

## 8.2 La virgola intelligente

Se viene caricato il pacchetto `babel` con l'opzione "`italian`", viene attivata automaticamente la virgola intelligente da esso definita. In questo modo sarà possibile scrivere agevolmente i numeri decimali con la virgola come separatore. Ricordiamoci che nel caso si voglia usare la virgola come simbolo di interpunzione in modalità matematica, è necessario lasciare uno spazio dopo di essa. Gli esempi proposti illustrano il funzionamento della virgola intelligente.

---

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

```
...
```

Il rapporto fra la circonferenza e il diametro è dato da

```
\[
  \pi=3,14159\ldots
\]
```

Siano poi  $A(3, 4)$  e  $B(-2, 1)$  gli estremi di un segmento\ldots

---

Il rapporto fra la circonferenza e il diametro è dato da

$$\pi = 3,14159\dots$$

Siano poi  $A(3, 4)$  e  $B(-2, 1)$  gli estremi di un segmento...

Insistiamo nel ribadire che se non vengono lasciati spazi dopo la virgola, si possono ottenere scritte errate:  $A(3,4)$  e  $B(-2,1)$  producono rispettivamente  $A(3,4)$  e  $B(-2,1)$ . Se però dopo la virgola viene inserito un carattere non numerico (ad esempio un meno), la spaziatura risulta comunque corretta.

Per disattivare e riattivare manualmente la virgola intelligente basta inserire nel testo le dichiarazioni `\NoIntelligentComma` e `\IntelligentComma`.

## 8.3 Soprassegno per gli archi

Viene definito il comando `\arc` che pone il soprassegno arrotondato per indicare gli archi.

---

Si consideri l'arco  $\arc{AB}$  e la corrispondente corda di misura  $\overline{AB}=15$ .

---

Si consideri l'arco  $\widehat{AB}$  e la corrispondente corda di misura  $\overline{AB} = 15$ .

Viene definita inoltre una variante `\vararc`, qualora si preferisca un soprassegno per gli archi più marcato.

---

Si consideri l'arco  $\vararc{AB}$  e la corrispondente corda di misura  $\overline{AB}=15$ .

---

Si consideri l'arco  $\widehat{AB}$  e la corrispondente corda di misura  $\overline{AB} = 15$ .

## 8.4 Unità di misura

Se viene caricato il pacchetto `babel` con l'opzione "italian", viene attivata automaticamente la dichiarazione `\setISOcompliance`, che consente di scrivere le unità di misura in modalità matematica tramite il comando `\unit`.

---

```
\usepackage[italian]{babel}
...
\begin{esercizi}
\item Un'automobile sta viaggiando a  $20\unit{m/s}$  lungo una strada
rettilenea. Ad un certo istante comincia ad aumentare la sua
velocità con accelerazione costante  $a=4,5\unit{m/s^2}$ . Quale
velocità avrà dopo  $10\unit{s}$ ?
\end{esercizi}
```

---

1. Un'automobile sta viaggiando a 20 m/s lungo una strada rettilinea. Ad un certo istante comincia ad aumentare la sua velocità con accelerazione costante  $a = 4,5 \text{ m/s}^2$ . Quale velocità avrà dopo 10 s?

In particolare, il simbolo dei gradi può essere inserito direttamente da tastiera<sup>5</sup>, mentre il simbolo  $\mu$  (prefisso di *micro* nelle unità di misura) è prodotto dal comando `\micro`.

---

L'angolo  $\alpha$  misura  $30^\circ$ . La temperatura della stanza è di  $22\unit{^\circ C}$ . Consideriamo una distanza di  $2,0\unit{\micro m}$ .

---

L'angolo  $\alpha$  misura  $30^\circ$ . La temperatura della stanza è di  $22^\circ\text{C}$ . Consideriamo una distanza di  $2,0\ \mu\text{m}$ .

Infine, segnaliamo che è possibile caricare il pacchetto `siunitx` per la gestione e la scrittura delle unità di misura. In tal caso, però, non deve essere *mai* usato nel documento il comando `\unit`, altrimenti L<sup>A</sup>T<sub>E</sub>X produrrebbe un errore.

Se viene selezionata una lingua diversa dall'italiano, il comando `\unit` non è attivo. In questo caso si può utilizzare il comando `\unitx`, che nei casi più semplici consente di ottenere le stesse funzionalità di `\unit`.

---

```
\begin{esercizi}\selectlanguage{english}
\item The Gravitational constant is approximately
```

<sup>5</sup>Ciò è reso possibile grazie al pacchetto `newunicodechar`.

```
$6.67\times 10^{-11}\unitx{N\cdot m^2/kg^2}$. \vf
\end{esercizio}
```

1. The Gravitational constant is approximately  $6.67 \times 10^{-11} \text{ N} \cdot \text{m}^2/\text{kg}^2$ .  T  F

## 8.5 Definizioni abbreviate

Elenchiamo nella tabella 3 alcune definizioni di comandi abbreviati.

Simbolo	L <sup>A</sup> T <sub>E</sub> X	verifica
$\delta$	<code>\delta</code>	<code>\del</code>
$\Delta$	<code>\Delta</code>	<code>\Del</code>
$\varepsilon$	<code>\varepsilon</code>	<code>\eps</code>
$\mathbb{N}$	<code>\mathbb{N}</code>	<code>\N</code>
$\mathbb{Z}$	<code>\mathbb{Z}</code>	<code>\Z</code>
$\mathbb{Q}$	<code>\mathbb{Q}</code>	<code>\Q</code>
$\mathbb{R}$	<code>\mathbb{R}</code>	<code>\R</code>

Tabella 3: Definizioni abbreviate di alcuni comandi.

I comandi `\phi` e `\theta` vengono ridefiniti per produrre  $\varphi$  e  $\vartheta$  (come `\varphi` e `\vartheta`). Per ripristinare i comandi originari basta dare le dichiarazioni `\restorephi` e `\restoretheta`.

Di default sono definiti `\phi` e `\theta`, ma si possono ripristinare `\restorephi\restoretheta` in modo che appaiano `\phi` e `\theta`.

Di default sono definiti  $\varphi$  e  $\vartheta$ , ma si possono ripristinare in modo che appaiano  $\phi$  e  $\theta$ .

Il valore assoluto si ottiene col comando `\abs`, definito seguendo le indicazioni della documentazione del pacchetto `mathtools`. Il suo utilizzo è mostrato di seguito.

```
\[
\abs{x} \quad \abs[\bigg]{\frac{x+2}{x-1}}
\]
```

$$|x| \quad \left| \frac{x+2}{x-1} \right|$$

## 9 Implementation

### 9.1 Class options

We define new conditionals for class options: `\if@blacktriangleright` is true by default and sets the `itemize` label symbol; `\if@intestrighe` and `\if@intestsemplice` are true if we choose the document heading “with lines” or “plain”, respectively; `\if@extrasize` is true if we set the font size to 14pt; `\if@senzagrazie` is true if we choose the sans serif font (cmbright package).

```
1 \newif\if@blacktriangleright
2 \newif\if@intestrighe
3 \newif\if@intestsemplice
4 \newif\if@extrasize
5 \newif\if@senzagrazie
```

Now, we define the class options and set `\@blacktrianglerighttrue` by default.

```
6 \DeclareOption{10pt}{\def\ver@ptsize{0}}
7 \DeclareOption{11pt}{\def\ver@ptsize{1}}
8 \DeclareOption{12pt}{\def\ver@ptsize{2}}
9 \DeclareOption{14pt}{\@extrasizetrue}
10 \DeclareOption{default}{\@blacktrianglerighttrue}
11 \DeclareOption{itembullet}{\@blacktrianglerightfalse}
12 \DeclareOption{intestazionerighe}{\@intestrighttrue}
13 \DeclareOption{intestazionesemplice}{\@intestsemplicetrue}
14 \DeclareOption{senzagrazie}{\@senzagrazietrue}
```

The option “11pt” is added by default.

```
15 \ExecuteOptions{default,11pt}
16 \ProcessOptions
```

If the option “14pt” is chosen, the conditional `\if@extrasize` is true and the `extarticle` class will be loaded with the options “a4paper” and “14pt”. Otherwise, the `article` class will be loaded with the option “a4paper” and the font size set by the user (default “11pt”).

```
17 \if@extrasize
18   \LoadClass[a4paper,14pt]{extarticle}
19 \else
20   \LoadClass[a4paper,1\ver@ptsize pt]{article}
21 \fi
```

Below is the list of packages loaded by the `verifica` class.

```
22 \RequirePackage[shortlabels,inline]{enumitem}
23 \RequirePackage{graphicx}
24 \RequirePackage{mathtools}
25 \RequirePackage{amssymb}
26 \RequirePackage{geometry}
27 \RequirePackage{nccmath}
28 \RequirePackage{multido}
29 \RequirePackage{setspace}
30 \RequirePackage{multicol}
31 \RequirePackage{gensymb}
32 \RequirePackage{newunicodechar}
33 \RequirePackage{textcomp}
34 \RequirePackage{tabto}
35 \RequirePackage{tabularx}
```

```
36 \RequirePackage[running]{lineno}
37 \RequirePackage{tikz}
```

Margins are set using the `geometry` package. We also set the page style to “empty”.

```
38 \geometry{top=1cm,bottom=1.5cm,left=1.5cm,right=1.5cm}
39 \pagestyle{empty}
```

If the option “`senzagrazie`” is chosen, then the `cmbright` package is loaded, so we have a sans serif font. The control sequence `\AtBeginDocument` causes the `cmbright` package to be loaded after all other packages.

```
40 \if@senzagrazie
41   \AtBeginDocument{%
42     \RequirePackage{cmbright}
43   }
44 \fi
```

The following lines contain various commands which must be placed at the beginning of the document, through the `\AtBeginDocument` command. If the `babel-italian` package is loaded, then `\setISOcompliance` and `\IntelligentComma` will be activated.

```
45 \AtBeginDocument{%
46   \@ifpackagewith{babel}{italian}{%
47     \setISOcompliance
48     \IntelligentComma
49   \relax}{}
```

If the `cmbright` package is loaded, the following declaration fixes the font for the bold-math mode, used in conjunction with the `bm` package. Otherwise, we will get the “poor man bold”.

```
50 \@ifpackageloaded{cmbright}{%
51   \SetSymbolFont{operators}{bold}{OT1}{cmbr}{b}{n}
52 \relax}{}
```

The `eurosym` and `bm` packages will be loaded after all the other packages (in particular, `eurosym` must be loaded after `babel`, and `bm` after `cmbright`).

```
53 \RequirePackage{eurosym}
54 \RequirePackage{bm}
```

`\restorephi` We redefine the `\phi` and `\theta` commands to get `\varphi` and `\vartheta`. We `\restoretheta` copy the original definitions to `\@tempvarphi` and `\@tempvartheta` for restoring them eventually through the `\restorephi` and `\restoretheta` commands.

```
55 \let\@tempvarphi\phi
56 \let\@tempvartheta\theta
57 \renewcommand{\phi}{\varphi}
58 \renewcommand{\theta}{\vartheta}
59 \newcommand{\restorephi}{\renewcommand{\phi}{\@tempvarphi}}
60 \newcommand{\restoretheta}{\renewcommand{\theta}{\@tempvartheta}}
61 }
```

`\abs` The next definition of the “absolute value” command is taken from the documentation of the `mathtools` package.

```
62 \DeclarePairedDelimiter\abs{\lvert}{\rvert}
```

## 9.2 The document heading

`\tipologia` The following lines are related to the creation of the document heading. First, we define the markup commands that pass their arguments to the internal macros. Every `\istituto` command is set to a default value, except `\@classe`, `\@tempo` and `\@lineanome`, which are undefined.

```

\classe
\tempo
\lineanome
63 \def\@tipologia{verifica}
64 \newcommand{\tipologia}[1]{\def\@tipologia{#1}}
65 \def\@disciplina{matematica}
66 \newcommand{\disciplina}[1]{\def\@disciplina{#1}}
67 \def\@istituto{Liceo ‘‘G. Novello’’ -- Codogno}
68 \newcommand{\istituto}[1]{\def\@istituto{#1}}
69 \def\@data{\today}
70 \newcommand{\data}[1]{\def\@data{#1}}
71 \let\@classe\undefined
72 \newcommand{\classe}[1]{\def\@classe{Classe-#1}}
73 \let\@tempo\undefined
74 \newcommand{\tempo}[1]{\def\@tempo{#1}}
75 \let\@lineanome\undefined
76 \newcommand{\lineanome}{\def\@lineanome{\relax}}

```

`\intestazionedefault` We define the default heading. It consists of two tabular environments, side by side, stored in the control sequences `\@tabellauno` and `\@tabelladue`. The first contains information about the discipline, the typology of the test, the institute name and the date. The second contains an empty space (which must be filled in by the candidate’s name) and the class (eventually empty), if defined. The length of the two tables are stored in `\@prima` and `\@seconda`. The length of `\@tabelladue` depends on the length of `\@tabellauno`. We use the `calc` package, which is loaded by the `mathtools` package, to determine the width of `\@tabellauno` (through the `\widthof` command). If `\tempo` is defined, the time available for the test will be printed. If `\lineanome` is defined, a dotted line for writing the candidate’s name will be printed.

```

77 \newlength{\@prima}
78 \newlength{\@seconda}
79 \def\intestazionedefault{%
80   \def\@tabellauno{%
81     \begin{tabular}{|c|}
82       \hline \[-.2cm]
83       \textbf{\MakeUppercase{\@tipologia} DI \MakeUppercase{\@disciplina}}\[-.2cm]
84       \textbf{\@istituto}\[-.2cm]
85       \textbf{\@data}\[-.2cm]
86       \hline
87     \end{tabular}}
88   \setlength{\@prima}{\widthof{\@tabellauno}}
89   \setlength{\@seconda}{\textwidth-\@prima-.7cm}
90   \ifdefined\@classe
91     \def\@@classe{\@classe}
92   \else
93     \def\@@classe{}
94   \fi
95   \ifdefined\@lineanome
96     \def\@@lineanome{\raisebox{-4pt}{\dotrule{\linewidth}}}

```

```

97 \else
98 \def\@lineanome{\relax}
99 \fi
100 \def\@tabelladue{%
101 \begin{tabular}{|p{\@seconda}|}
102 \hline \\\[-.2cm]
103 Cognome e nome \\\[.2cm]
104 \@lineanome \\\[.2cm]
105 \@@classe \\\[.2cm]
106 \hline
107 \end{tabular}}
108 \ifdefined\@tempo
109 \def\tempo@ad{\begin{center}%
110 \textit{Tempo a disposizione: \@tempo}\end{center}}
111 \else
112 \def\tempo@ad{\vspace{.5cm}}
113 \fi
114 \noindent\@tabellauno\hfill\@tabelladue\tempo@ad}

```

`\intestazionerighe` We define the second type of heading: “intestazione con righe” (“with lines”). It puts the information about the document between two horizontal lines. If `\lineanome` is defined, a field for writing the candidate’s name will be printed.

```

115 \def\intestazionerighe{%
116 \ifdefined\@classe
117 \def\@@classe{\@classe}
118 \else
119 \def\@@classe{}
120 \fi
121 \ifdefined\@lineanome
122 \def\@@lineanome{%
123 \noindent {\small Cognome e nome} \dotrule{.3\textwidth}\par\vskip6pt}
124 \else
125 \def\@@lineanome{\relax}
126 \fi
127 \ifdefined\@tempo
128 \def\tempo@ad{\centering%
129 \textit{Tempo a disposizione: \@tempo}\par}\vspace{.5em}}
130 \else
131 \def\tempo@ad{\vspace{.1em}}
132 \fi
133 \@@lineanome
134 \null\vspace{-1cm}
135 \noindent\hrulefill \\\
136 \noindent\makebox[.3\linewidth][l]{\small\@istituto} \hfill%
137 \makebox[.3\linewidth]{\@@classe} \hfill%
138 \makebox[.3\linewidth][r]{\small\@data}%
139 \begin{center}
140 \textbf{\MakeUppercase{\@tipologia} DI \MakeUppercase{\@disciplina}}\\\[-.5em]
141 \hrulefill
142 \end{center}
143 \tempo@ad}

```

`\intestazione semplice` We define the third type of heading: “intestazione semplice” (“plain”). It is similar to the “intestazione con righe”, but without the two horizontal lines.

```

144 \def\intestazionesemplice{%
145 \ifdefined\@classe
146   \def\@@classe{\@classe}
147 \else
148   \def\@@classe{}
149 \fi
150 \ifdefined\@lineanome
151   \def\@@lineanome{%
152     \noindent {\small Cognome e nome} \dotrule{.3\textwidth}\par\vskip4pt}
153 \else
154   \def\@@lineanome{\relax}
155 \fi
156 \ifdefined\@tempo
157   \def\tempo@ad{\centering%
158     \textit{Tempo a disposizione: \@tempo}\par}\vspace{.5em}}
159 \else
160   \def\tempo@ad{\vspace{.1em}}
161 \fi
162 \@@lineanome
163 \noindent\makebox[.3\linewidth][l]{\small\@istituto} \hfill%
164 \makebox[.3\linewidth]{\@@classe} \hfill%
165 \makebox[.3\linewidth][r]{\small\@data}%
166 \begin{center}
167   \textbf{\MakeUppercase{\@tipologia} DI \MakeUppercase{\@disciplina}}\[-.5em]
168 \end{center}
169 \tempo@ad}

```

`\intestazione` The `\intestazione` command generates the document heading.

```

170 \if@intestrighe
171   \def\intestazione{\intestazionerighe}
172 \else
173   \if@intestsemplice
174     \def\intestazione{\intestazionesemplice}
175   \else
176     \def\intestazione{\intestazionedefault}
177   \fi
178 \fi

```

### 9.3 New commands

`\tsa` Here are some definition of new commands. Thanks to the `newunicodechar` package, we can type the character  $^{\circ}$  to get the corresponding symbol directly. The commands `\tsa` and `\tso` produce the ordinal number symbols  $^{\text{a}}$  and  $^{\text{o}}$ , followed by a thin space.

```

179 \newunicodechar{^{\circ}}{\ensuremath{\sim\circ}}
180 \newcommand{\tsa}{\textsuperscript{a}\,}
181 \newcommand{\tso}{\textsuperscript{o}\,}

```

`\eps` The following lines define some useful abbreviation for some frequently used `\del` commands.

```

\Del
\R 182 \newcommand{\eps}{\varepsilon}
\Q 183 \newcommand{\del}{\delta}
\Z
\N
\dss

```

```

184 \newcommand{\Del}{\Delta}
185 \newcommand{\R}{\mathbb{R}}
186 \newcommand{\Q}{\mathbb{Q}}
187 \newcommand{\Z}{\mathbb{Z}}
188 \newcommand{\N}{\mathbb{N}}
189 \newcommand{\dss}[1]{\displaystyle{#1}}

```

`\arc` The following code is taken from <https://tex.stackexchange.com/questions/96680#284787> (thanks to egreg). It defines a symbol to denote an arc.

```

190 \DeclareFontFamily{U}{tipa}{}
191 \DeclareFontShape{U}{tipa}{m}{n}{<->tipa10}{}
192 \newcommand{\arc@char}{\usefont{U}{tipa}{m}{n}\symbol{62}}
193 \newcommand{\arc}[1]{\mathpalette\arc@arc{#1}}
194 \newcommand{\arc@arc}[2]{%
195   \sbox0{\m@th#1#2}%
196   \vbox{%
197     \hbox{\resizebox{\wd0}{\height}{\arc@char}}
198     \nointerlineskip
199     \box0
200   }
201 }

```

`\vararc` Here is a variant of the same symbol (more marked).

```

202 \DeclareFontFamily{OMX}{yhex}{}
203 \DeclareFontShape{OMX}{yhex}{m}{n}{<->yhcmex10}{}
204 \DeclareSymbolFont{yhlargesymbols}{OMX}{yhex}{m}{n}
205 \DeclareMathAccent{\vararc}{\mathord}{yhlargesymbols}{"F3}

```

`\ddfrac` We define the `\ddfrac` command for “double” fractions (fractions of fractions),  
`\numfrac` with a slightly longer line. The `\numfrac` and `\denfrac` commands produce correct  
`\denfrac` vertical spacing in double fractions.

```

206 \newcommand\ddfrac[2]{\frac{\,\displaystyle #1\,}{\displaystyle #2\,}}
207 \newcommand\numfrac[2]{\frac{#1}{#2\mathstrut}}
208 \newcommand\denfrac[2]{\frac{\mathstrut#1}{#2}}

```

`\unitx` If `babel-italian` is not loaded or a different language is temporarily set for the document, the `\unit` command will not be available. In this case, the `\unitx` command can be used in place of `\unit`: the `\unitx` command is roughly similar to `\unit`.

```

209 \newcommand{\unitx}[1]{\ensuremath{\,\mathrm{#1}}}

```

By default, `\if@blacktriangleright` is true and the `itemize` label symbol is  $\blacktriangleright$ . If the option “`itembullet`” is chosen, `\if@blacktriangleright` is false and the `itemize` label symbol is  $\bullet$ .

```

210 \if@blacktriangleright
211   \def\labelitemi{\small $\blacktriangleright$}
212 \fi

```

## 9.4 Test environments

`\labelbox` We use the `enumitem` package to define the main environment `esercizi` and various  
`\biglabelbox` lists for multiple choice tests. By default, the label for the multiple choice test, stored in `\@crisp`, consists of an alphabetical counter inside a squared framed box. There are two types of boxes: normal (`\labelbox`, used for multiple choice tests) and big (`\biglabelbox`, used for true/false tests). If the option “`senzagrazie`” is set, the default style for the alphabetic counter is upright, otherwise is italic. If the option “`14pt`” is set, the size of these boxes increases.

```

213 \if@senzagrazie
214   \def\@labelstyle{\textup}
215 \else
216   \def\@labelstyle{\textit}
217 \fi
218 \newcommand{\labelbox}[1]{\fbox{\parbox[] [.2cm] [c]{.2cm}{#1}}}
219 \newcommand{\biglabelbox}[1]{\fbox{\parbox[b] [.25cm] [c]{.25cm}{#1}}}
220 \if@extrasize
221   \renewcommand{\labelbox}[1]{\fbox{\parbox[] [.3cm] [c]{.3cm}{#1}}}
222   \renewcommand{\biglabelbox}[1]{\fbox{\parbox[b] [.35cm] [c]{.35cm}{#1}}}
223 \fi
224 \def\@crisp{\labelbox{\@labelstyle{\alph*}}}
```

`esercizi` (*env.*) We define the new lists `esercizi`, `test` and `test-orizz`. The last one is a “horizontal” list, that is, it’s an inline `enumitem` list. The `\labeltest` command allows  
`test` (*env.*) us to change the label style of all test environments, even in the “`shortlabels`”  
`test-orizz` (*env.*) `\labeltest` `enumitem` mode.

```

225 \newlist{esercizi}{enumerate}{1}
226 \setlist[esercizi]{\bfseries 1.,itemsep=2em,leftmargin=*}
227 \newlist{test}{enumerate}{1}
228 \setlist[test]{\@crisp,leftmargin=*}
229 \newlist{test-orizz}{enumerate*}{1}
230 \setlist[test-orizz]{%
231   \@crisp,before=\hspace{1em},itemjoin=\hspace{1.2em},afterlabel=~}
232 \newcommand{\labeltest}[1]{%
233   \renewlist{test}{enumerate}{1}
234   \setlist[test]{#1,leftmargin=*}
235   \renewlist{test-orizz}{enumerate*}{1}
236   \setlist[test-orizz]{%
237     #1,before=\hspace{1em},itemjoin=\hspace{1.2em},afterlabel=~}
238 }
```

`test-orizz-newline` (*env.*) The `test-orizz-newline` environment is similar to `test-orizz`, but multiple choices start on a new line.

```

239 \newenvironment{test-orizz-newline}[1] []{%
240   \par\begin{test-orizz}[before=\noindent,#1]{\end{test-orizz}}
```

## 9.5 True/False tools

`\vf` We define tools for creating true/false tests. If the selected language is English, or the `babel` package is not loaded, the `\vf` command generates the true/false boxes `T` `F`. Otherwise, it generates `V` `F` (especially when the selected language is Italian). The boxes will be placed at the end of the current line. The `\vf` command is right for short questions (shorter than the line width).

```

241 \newcommand{\vf}{-%
242   \iflanguage{english}{-%
243     \hfill{\small\biglabelbox{T}~\biglabelbox{F}}}{-%
244     \hfill{\small\biglabelbox{V}~\biglabelbox{F}}}{-%
245   }

```

Now, we define a more complex macro for creating long true/false questions.

`\fattorevf` The control sequence `\fattorevf` contains the scale factor that determines the line break point. It is set by default to 0.75. It can be changed using the `\fattorevf{<scale factor>}` command, which takes as an argument a number from 0 to 1 and overwrites the previous value of `\@fattorevf`. If the argument is not a number between 0 and 1, we will get an error message.

```

246 \def\@fattorevf{.75}
247 \def\@fattorevferror{%
248   \ClassError{verifica}{-%
249     \protect\fattorevf\space e fattorevf devono avere come argomento
250     un\MessageBreak numero compreso tra 0 e 1}{-%
251     Riprova scrivendo \protect\fattorevf{.85} o fattorevf=.85
252     o qualcosa di simile. Ciao ;) \MessageBreak}{-%
253   }
254 \newcommand{\fattorevf}[1]{-%
255   \ifdimOpt<#1pt\relax
256     \ifdim1pt>#1pt\relax
257       \def\@fattorevf{#1}
258     \else
259       \@fattorevferror
260     \fi
261   \else
262     \@fattorevferror
263   \fi
264 }

```

`\CalcNumber` In the following lines we define the internal macro `\CalcNumber`, which counts the number of lines of a text. The text is put in a minipage of the same width as the current line, which in turn is placed in the `\@boxcountlines` savebox. The `lineno` package allows us to determine the number of the lines of the text, which is stored in the `mylines` counter.

This macro is taken from <https://tex.stackexchange.com/questions/402779#406791> (thanks to user121799).

```

265 \newcounter{mylines}
266 \newsavebox{\@boxcountlines}
267 \newcommand{\CalcNumber}[2]{-%
268 \savebox{\@boxcountlines}{-%
269   \begin{minipage}{\linewidth}\linenumbers*
270     \begin{internallinenumbers}
271       #1\par
272       \setcounter{mylines}{\thelinenumber}\addtocounter{mylines}{-1}
273     \end{internallinenumbers}
274   \end{minipage}}
275 \let#2=\themylines
276 }

```

`\Repeat` The next macro is defined through the `multido` package. It causes the repetition of the command that follows it as many times as specified by the argument.

```
277 \newcommand{\Repeat}{\multido{\i=1+1}}
```

`\verofalso` The macro `\verofalso` [*scale factor*] {*text*} is defined to handle a long text in a true/false test. The mandatory argument is the text of a true/false question and the optional one is the scale factor for the line width (set by default by `\@fatterevf`). The macro puts the text in a `tabularx` environment (`tabularx` package), within a cell of width equal to the resized line width, and passes it to the `\CalcNumber` macro, which counts the number of lines and stores this value in the counter `\myx`. Since this counting takes place in a cell, the current `\linewidth` is equal to the cell width. The `\numrighe` counter contains the number of `\newline` required to correctly position the true/false boxes within the second `tabularx` cell, at the same level as the last line of the question.

```
278 \newcounter{numrighe}
279 \newcommand{\verofalso}[2][\@fatterevf]{%
280   \renewcommand{\tabcolsep}{0em}%
281   \begin{tabularx}{\linewidth}[b]{p{#1\linewidth}X}%
282     {#2}\CalcNumber{#2}{\myx}\setcounter{numrighe}{\myx-1}%
283     & \phantom{e}\Repeat{\value{numrighe}}\newline\phantom{e}\vf \\
284   \end{tabularx}
285 }
```

`test-verofalso` (*env.*) We define the `test-verofalso` environment. First, we define the `test-vf` list through the `enumitem` package and the “`fatterevf`” parameter, which sets the value of the `\fatterevf` command. Second, we define the `\vfitem` command, which has the text of the item as an argument.

```
286 \enitkv@key{}{\fatterevf}{\fatterevf{#1}}
287 \newlist{test-vf}{enumerate}{1}
288 \setlist[test-vf]{label=\alph*}
289 \newcommand{\vfitem}[1]{\item\verofalso{#1}}
290 \newenvironment{test-verofalso}[1][ ]{%
291   \begin{test-vf}[#1]\end{test-vf}}
```

## 9.6 Dotted lines

`\dotrule` The `\dotrule` {*length*} command inserts a dotted line of length equal to its argument; the `\dotword` {*word*} command inserts a dotted line that is twice the width of the word specified as an argument.

```
292 \newcommand{\@leadersbox}{.45em}
293 \newcommand{\@tratto}{.}
294 \newcommand{\@completeleaders}{%
295   \leavevmode\leaders\hbox to \@leadersbox{\hss\@tratto\hss}\hfill\kern0pt}
296 \newcommand{\dotrule}[1]{%
297   \parbox[t]{#1}{\@completeleaders}}
298 \newcommand{\dotword}[1]{%
299   \parbox[t]{\widthof{#1} * \real{2}}{\@completeleaders}}
```

`\riga` The `\riga` {*number*} command inserts dotted lines with double vertical spacing, as many as specified by the argument. It is defined through the `multido` and `setspace` packages.

```

300 \newcommand{@rigaleaders}{%
301   \leavevmode\leaders\hbox to \@leadersbox{\hss\@tratto\hss}\hfill\kern0pt\par}
302 \newcommand{riga}[1]{%
303   \doublespacing\Repeat{#1}{\noindent@rigaleaders}\singlespacing}

```

## 9.7 Grids of squares

`\drawgrid` The `\drawgrid[square side][height]` command draws a grid of squares with the specified height and the width equal to the current line length. The default square size is 0.5 cm, but it can be changed via the optional argument. If necessary, the height of the grid is rounded down to the nearest multiple of the square size that does not exceed the specified height.

```

304 \newlength{gridsize}
305 \newlength{gridwidth}
306 \newlength{desiredHeight}
307 \newlength{gridheight}
308 \newcommand{drawgrid}[2][.5cm]{%
309   \setlength{gridsize}{#1}
310   \pgfmathsetlength{gridwidth}{gridsize * floor(\linewidth/gridsize)}
311   \setlength{desiredHeight}{#2}
312   \pgfmathsetlength{gridheight}{gridsize * floor(\desiredHeight/gridsize)}
313   \begin{center}
314     \begin{tikzpicture}
315       \begin{scope}[xshift=\dimexpr(\linewidth - \gridwidth)/2\relax]
316         \draw[step=gridsize,gray!80,semithick]
317           (0,0) grid (\gridwidth,\gridheight);
318       \end{scope}
319     \end{tikzpicture}
320   \end{center}
321 }

```

## 9.8 Scoring tools

`\punti` The `\punti[additional text][score]` command places the score at the end of the current line. If there is not enough space at the end of the current line, the score will be automatically positioned at the end of a new line. If the optional argument is defined, an additional text will be written after the score, automatically separated by a non-breaking space. If the optional argument is missing, this non-breaking space disappears: this feature is made possibile by the macro described at <https://tex.stackexchange.com/questions/217757#217763> (thanks to David Carlisle).

```

322 \newcommand{@punteggio}{Punti}
323 \newcommand{punti}[2][@nil]{%
324   \hfill\vphantom{,}\nolinebreak[1]\null~\hfill%
325   [\textsc{@punteggio-#2}]
326   \def\tmp{#1}%
327   \ifx\tmp@nnil
328     \relax
329   \else
330     ~\mbox{#1}%
331   \fi}]%
332 }

```

`\puntiman` Unfortunately, there is a bug: if the first word of the line where the score has to be printed is part of a hyphenated word, the score will not be printed on the same line, but on the next (empty). To solve this issue “manually”, the `\puntiman` command can be used instead of `\punti`.

```

333 \newcommand{\puntiman}[2][\@nil]{%
334   \hfill%
335   [\textsc{\@punteggio~#2%
336     \def\tmp{#1}%
337     \ifx\tmp\@nnil
338       \relax
339     \else
340       ~\mbox{#1}%
341     \fi}}%
342 }

```

`\puntiadestra` This command can be placed anywhere in the document and simply prints the word “Punti” after the score.

```

343 \newcommand{\punti@dx}[2][\@nil]{%
344   \hfill\vphantom{,}\nolinebreak[1]\null~\hfill%
345   [\textsc{#2~\@punteggio%
346     \def\tmp{#1}%
347     \ifx\tmp\@nnil
348       \relax
349     \else
350       ~\mbox{#1}%
351     \fi}}%
352 }
353 \newcommand{\puntiman@dx}[2][\@nil]{%
354   \hfill%
355   [\textsc{#2~\@punteggio%
356     \def\tmp{#1}%
357     \ifx\tmp\@nnil
358       \relax
359     \else
360       ~\mbox{#1}%
361     \fi}}%
362 }
363 \def\puntiadestra{%
364   \renewcommand{\punti}{\punti@dx}
365   \renewcommand{\puntiman}{\puntiman@dx}
366 }

```

`\punti` This command can also be placed anywhere in the document and replaces the word “Punti” with the abbreviation “Pt.”

```

367 \def\punti{Pt.}
368 \renewcommand{\@punteggio}{Pt.}

```

`\totpunti` The `\totpunti[calculation]` command creates a `tabular` environment at the end of the current page, which is used to write the total score and calculate the final grade. The optional calculation is set by default to  $/10+1$  (the one I usually adopt), but it can be changed as desired. The optional argument will be automatically set in math mode.

```

369 \newcommand{\totpunti}[1][ /10 + 1 ]{%

```

```

370 \vfill
371 \begin{flushright}
372 {\renewcommand{\arraystretch}{1.5}
373 \begin{tabular}{|cc|}
374 \hline
375 & {\hspace{-7pt}\textbf{VOTO}} \\\
376 {\textbf{Totale punti} \fbox{\phantom{\LARGE MM}} $\bm{\#1=}$}%
377 & \hspace{-7pt}\fbox{\phantom{\Huge MM}} \\\
378 \hline
379 \end{tabular}}
380 \end{flushright}}

```

`\vartotpunti` This command is similar to `\totpunti`, but it doesn't place the final score `tabular` environment at the end of the page.

```

381 \newcommand{\vartotpunti}[1][10 + 1]{%
382 \begin{flushright}
383 {\renewcommand{\arraystretch}{1.5}
384 \begin{tabular}{|cc|}
385 \hline
386 & {\hspace{-7pt}\textbf{VOTO}} \\\
387 {\textbf{Totale punti} \fbox{\phantom{\LARGE MM}} $\bm{\#1=}$}%
388 & \hspace{-7pt}\fbox{\phantom{\Huge MM}} \\\
389 \hline
390 \end{tabular}}
391 \end{flushright}}

```

## 10 Change History

v1.0	General: First public release . . . . .	<b>1</b>	score at the end of the line when the <code>\punti</code> command fails. . . . .	<b>34</b>
v1.1	<code>\lineanome</code> : Added the <code>\lineanome</code> command, which draws a dotted line for writing the candidate's name. . . . .	<b>27</b>	v1.3	General: Fixed a bug related to the <code>eurosym</code> package, which must be loaded after <code>babel</code> . . . . .
	<code>test-verofalso</code> : Added the <code>test-verofalso</code> environment	<b>33</b>	v2.0	<code>\drawgrid</code> : Added the <code>\drawgrid</code> command, which draws a grid of squares. . . . .
v1.2	<code>\puntiman</code> : Added the <code>\puntiman</code> command, which prints the			<b>34</b>

## 11 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

<b>Symbols</b>	<b>137, 146, 148, 164</b>	<b>133, 151, 154, 162</b>
<code>\@@classe</code> . . . . .	<b>91,</b>	<code>\@@lineanome</code> . . .
<b>93, 105, 117, 119,</b>	<b>98, 104, 122, 125,</b>	<code>\@blacktrianglerightfalse</code>
		. . . . .

<code>\@blacktrianglerighttrue</code>	..... 10	<code>\Del</code> .....	182, 184	N	
<code>\@boxcountlines</code>	266, 268	<code>\del</code> .....	182, 183	<code>\N</code> .....	182, 188
<code>\@classe</code>	71, 72, 90, 91, 116, 117, 145, 146	<code>\denfrac</code> .....	206	<code>\numfrac</code> .....	206
<code>\@completeleaders</code>	..... 294, 297, 299	<code>\desiredHeight</code> .....	306, 311, 312	P	
<code>\@data</code>	69, 70, 85, 138, 165	<code>\dimexpr</code> .....	315	<code>\pgfmathsetlength</code> ..	..... 310, 312
<code>\@disciplina</code>	..... 65, 66, 83, 140, 167	<code>\disciplina</code> .....	63, 66	<code>\punti</code> .....	322, 364
<code>\@extrasizetrue</code>	..... 9	<code>\dotrule</code> 96, 123, 152, 292		<code>\punti@dx</code> .....	343, 364
<code>\@fattorevf</code>	246, 257, 279	<code>\dotword</code> .....	292	<code>\puntiadestra</code> .....	343
<code>\@fattorevferror</code>	..... 247, 259, 262	<code>\draw</code> .....	316	<code>\puntiman</code> .....	333, 365
<code>\@intestrihgetrue</code>	.. 12	<code>\drawgrid</code> .....	304	<code>\puntiman@dx</code> ..	353, 365
<code>\@intestsemplcetrue</code>	13	<code>\dss</code> .....	182, 189	<code>\punti@pt</code> .....	367
<code>\@istituto</code>	..... 67, 68, 84, 136, 163	E		Q	
<code>\@labelstyle</code>	214, 216, 224	environments:		<code>\Q</code> .....	182, 186
<code>\@leadersbox</code>	292, 295, 301	<code>esercizi</code> .....	225	R	
<code>\@lineanome</code>	..... 75, 76, 95, 121, 150	<code>test</code> .....	225	<code>\R</code> .....	182, 185
<code>\@prima</code>	..... 77, 88, 89	<code>test-orizz</code> .....	225	<code>\raisebox</code> .....	96
<code>\@punteggio</code>	322, 325, 335, 345, 355, 368	<code>test-orizz-newline</code> ..	239	<code>\Repeat</code> ...	277, 283, 303
<code>\@rigaleaders</code>	300, 303	<code>test-verofalso</code> ..	286	<code>\restorephi</code> .....	55, 59
<code>\@risp</code>	..... 224, 228, 231	<code>\eps</code> .....	182, 182	<code>\restoretheta</code> ...	55, 60
<code>\@seconda</code>	.. 78, 89, 101	<code>esercizi (env.)</code> .....	225	<code>\riga</code> .....	300
<code>\@senzagrazietrue</code>	.. 14	F		T	
<code>\@tabelladue</code>	.. 100, 114	<code>\fattorevf</code> .....	246, 286	<code>\tempo</code> .....	63, 74
<code>\@tabellauno</code>	80, 88, 114	G		<code>\tempo@ad</code> ..	109, 112, 114, 128, 131, 143, 157, 160, 169
<code>\@tempo</code>	..... 73, 74, 108, 110, 127, 129, 156, 158	<code>\gridheight</code> 307, 312, 317		<code>test (env.)</code> .....	225
<code>\@tempvarphi</code>	..... 55, 59	<code>\gridsize</code> .....	304, 309, 310, 312, 316	<code>test-orizz (env.)</code> ...	225
<code>\@tempvartheta</code>	.. 56, 60	<code>\gridwidth</code> .....	305, 310, 315, 317	<code>test-orizz-newline</code>	(env.) .....
<code>\@tipologia</code>	..... 63, 64, 83, 140, 167	I		<code>test-verofalso (env.)</code>	286
<code>\@tratto</code>	.. 293, 295, 301	<code>\if@blacktriangleright</code>	..... 1, 210	<code>\thelinenumber</code> .....	272
A		<code>\if@extrasize</code>	4, 17, 220	<code>\themylines</code> .....	275
<code>\abs</code> .....	62, 62	<code>\if@intestrighe</code>	2, 170	<code>\tipologia</code> .....	63, 64
<code>\arc</code> .....	190, 193	<code>\if@intestsemplce</code>	3, 173	<code>\totpunti</code> .....	369
<code>\arc@arc</code> .....	193, 194	<code>\if@senzagrazie</code>	5, 40, 213	<code>\tsa</code> .....	179, 180
<code>\arc@char</code> .....	192, 197	<code>\intestazione</code> .....	170	<code>\tso</code> .....	179, 181
B		<code>\intestazionedefault</code>	..... 77, 176	U	
<code>\biglabelbox</code>	213, 243, 244	<code>\intestazionerighe</code>	..... 115, 171	<code>\unitx</code> .....	209, 209
C		<code>\intestazioneemplce</code>	..... 144, 174	V	
<code>\CalcNumber</code> ...	265, 282	<code>\istituto</code> .....	63, 68	<code>\vararc</code> .....	202, 205
<code>\classe</code> .....	63, 72	L		<code>\vartotpunti</code> .....	381
D		<code>\labelbox</code> .....	213	<code>\ver@ptsiz</code> ..	6, 7, 8, 20
<code>\data</code> .....	63, 70	<code>\labeltest</code> .....	225	<code>\verofalso</code> .....	278, 289
<code>\ddfrac</code> .....	206	<code>\lineanome</code> .....	63, 76	<code>\vf</code> .....	241, 241, 283
		M		<code>\vfitem</code> .....	289
		<code>\myx</code> .....	282	<code>\vskip</code> .....	123, 152
		Z		<code>\Z</code> .....	182, 187