

La classe **verifica**

Riccardo Dossena*

19 gennaio 2020

Abstract

This L^AT_EX 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

1	Introduzione	2
2	Licenza	2
3	Come usare la classe verifica	2
3.1	Opzioni di classe	3
4	L'intestazione della verifica	3
4.1	Altri tipi di intestazione	5
4.2	Il nominativo del candidato	6
5	Gli ambienti	7
5.1	L'ambiente esercizi	7
5.2	Sottoelenchi dell'ambiente esercizi	8
5.3	Sottoelenchi puntati	9
5.4	Test a risposta multipla	10
5.5	Scrivere su più colonne	12
6	Strumenti	13
6.1	Vero o falso	13
6.2	Righe per scrivere le risposte	16
6.3	Completamenti	16
7	Punteggio	18
7.1	Comandi per inserire il punteggio	18
7.2	Varianti per il punteggio	19
7.3	Il riquadro del punteggio totale	20

*E-mail: riccardo.dossena@gmail.com

8	Miscellanea	20
8.1	Le frazioni doppie	20
8.2	La virgola intelligente	21
8.3	Soprassegno per gli archi	22
8.4	Unità di misura	22
8.5	Definizioni abbreviate	23
9	Implementation	25
9.1	Class options	25
9.2	The document heading	27
9.3	New commands	29
9.4	Test environments	31
9.5	True/False tools	31
9.6	Dotted lines	33
9.7	Scoring tools	34
10	Change History	35
11	Index	35

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 \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 \LaTeX di questa classe è rilasciato sotto la \LaTeX Project Public License, v1.3.

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 \TeX ¹. Per iniziare a editare un documento si può digitare direttamente il file `verifica-template.tex`, che contiene alcuni comandi già impostati.

¹Si vedano, nel caso, le \TeX FAQ alla pagina <https://texfaq.org/FAQ-inst-wlcf>.

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.

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

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

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

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

La classe `verifica` fornisce inoltre i comandi `\tsa` e `\tso`² che generano rispettivamente i simboli ^a e ^o 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}
\disciplina{fisica}
\istituto{Liceo "F. Indovino" -- Roma}
\classe{3\tsa C}
\data{12 novembre 2019}

\intestazione
```

VERIFICA ORALE DI FISICA Liceo "F. Indovino" – Roma 12 novembre 2019	Cognome e nome Classe 3 ^a 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
```

VERIFICA DI MATEMATICA Liceo "G. Novello" – Codogno 19 gennaio 2020	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.

²Abbreviazioni di `a` e `o`.

```
\begin{document}
```

```
\intestazione
```

VERIFICA DI MATEMATICA Liceo “G. Novello” – Codogno 19 gennaio 2020	Cognome e nome
--	----------------

4.1 Altri tipi di intestazione

È possibile usare altri due stili di intestazione tramite i comandi `\intestazionesemplice` e `\intestazionerighe`. 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}
```

```
\intestazionesemplice
```

Liceo “G. Novello” – Codogno	Classe 3 ^a C	12 novembre 2019
PROVA PER COMPETENZE DI MATEMATICA		

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 ^a B	12 novembre 2019
VERIFICA ORALE DI MATEMATICA		
<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 `\intestazione righe`, è possibile impostare un’opzione di classe. Scrivendo infatti

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

oppure

```
\documentclass[intestazione righe]{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 ^a B	12 novembre 2019
VERIFICA ORALE DI MATEMATICA		
<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 ^a B	12 novembre 2019
VERIFICA ORALE DI MATEMATICA		
<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.

VERIFICA ORALE DI MATEMATICA Liceo “G. Novello” – Codogno 12 novembre 2019	Cognome e nome Classe 2 ^a 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 2em.

```

\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 1em (anziché di 2em) 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=*[
    \item $\dss{\frac{x+1}{x-2}+4=\frac{x}{x-1}}$

    \item $\dss{\sin^2 x + \cos x = \cos 2x}$

    \item $\dss{\ln (x+1)=\ln 5 + \log_2 x}$
  \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"> • A. (numerazione alfabetica in maiuscolo con punto) • i. (numerazione romana in minuscolo con punto) • <code>\bfseries 1)</code> (numerazione in grassetto con parentesi)
<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[itembullet]{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 $1 + \sin x$? ☐ $\sin x$; ☐ $\cos x$;
☐ $-\sin x$; ☐ $-\cos x$; ☐ 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 $1 + \sin x$? ☐ $\sin x$; ☐ $\cos x$; ☐ $-\sin x$; ☐ $-\cos x$; ☐ 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`³.

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 $\langle label \rangle$ 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 *Giochi di Archimede*.

³Consultare, 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}{\langle numero di colonne \rangle}
...

```

```
\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=0]
    \item l'accelerazione media dell'automobile;
    \item lo spazio percorso in fase di accelerazione.
  \end{itemize}
\end{esercizi}
\end{multicols}}
```

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

- ☐ $\sin x$;
- ☐ $\cos x$;
- ☐ $-\sin x$;
- ☐ $-\cos x$;
- ☐ nessuna delle precedenti.

2. 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.

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 ☐ ☐. 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>}`⁴.

```
\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}
```

⁴La 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{esercizi}
\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{esercizi}
```

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}
```

1. Enuncia il teorema di Pitagora.

.....
.....
.....

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}
```

1. Enuncia il teorema di Pitagora.

.....
.....

6.3 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}
```

- | |
|---|
| <ol style="list-style-type: none">1. Data la retta $y = 2x + 1$, scrivi l'equazione di una sua perpendicolare. [PUNTI 10]2. Data la retta $y = 3x - 1$, calcola l'equazione della retta per A ad essa parallela. [PUNTI 10]3. 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] |
|---|

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}
```

- | |
|---|
| <ol style="list-style-type: none">1. Risolvi le seguenti equazioni. [PUNTI 10 PER OGNI EQUAZIONE]<ol style="list-style-type: none">a) $x^2 - 3x = 1$b) $2x + 1 = 4x^2 - 3$ |
|---|

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
    \[
      x=-\frac{1}{2}x^2-x+1 \tag*{\punti{5}}
    \]
\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 `\punctipt`, che sostituisce alla scritta “PUNTI” l'abbreviazione “PT.”

```
\punctipt
\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\punctipt
\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 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:

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

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 <input type="text"/>	$/12 + 2 =$ <input type="text"/>

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 \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`.

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

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

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. Ricordiamo 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
      rettilinea. 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 tastiera⁵, mentre il simbolo μ (prefisso di *micro* nelle unità di misura) è prodotto dal comando `\micro`.

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

L'angolo α misura 30° . La temperatura della stanza è di 22°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 \LaTeX 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
 $6.67 \times 10^{-11} \text{ N} \cdot \text{m}^2/\text{kg}^2$ . \vf
\end{esercizi}
```

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	\LaTeX	verifica
δ	<code>\delta</code>	<code>\del</code>
Δ	<code>\Delta</code>	<code>\Del</code>
ε	<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.

⁵Ciò è reso possibile grazie al pacchetto `newunicodechar`.

I comandi `\phi` e `\theta` vengono ridefiniti per produrre φ e ϑ (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 φ e ϑ , ma si possono ripristinare in modo che appaiano ϕ e θ .

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

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

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

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{eurosym}
35 \RequirePackage{tabto}
```

```

36 \RequirePackage{tabularx}
37 \RequirePackage[running]{lineno}

```

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 `bm` package will be loaded after all the other packages (in particular after `cmbright`).

```

53 \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.

```

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

```

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

```

61 \DeclarePairedDelimiter\abs{\lvert}{\rvert}

```

9.2 The document heading

<code>\tipologia</code> <code>\disciplina</code> <code>\istituto</code> <code>\data</code>	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 command is set to a default value, except <code>\@classe</code> , <code>\@tempo</code> and <code>\@lineanome</code> , which are undefined.
---	--

```

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

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



```

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

```

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

```

169 \if@intestrighe
170 \def\intestazione{\intestazionerighe}
171 \else
172 \if@intestsemplice
173 \def\intestazione{\intestazioneimplice}
174 \else
175 \def\intestazione{\intestationedefault}
176 \fi
177 \fi

```

9.3 New commands

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

```

178 \newunicodechar{◦}{\ensuremath{\sim\circ}}
179 \newcommand{\tsa}{\textsuperscript{a}\,}
180 \newcommand{\tso}{\textsuperscript{◦}\,}

```

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

```

\Del
181 \newcommand{\eps}{\varepsilon}
182 \newcommand{\del}{\delta}

```

`\Z`

`\N`

`\dss`

```

183 \newcommand{\Del}{\Delta}
184 \newcommand{\R}{\mathbb{R}}
185 \newcommand{\Q}{\mathbb{Q}}
186 \newcommand{\Z}{\mathbb{Z}}
187 \newcommand{\N}{\mathbb{N}}
188 \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.

```

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

```

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

```

201 \DeclareFontFamily{OMX}{yhhex}{}
202 \DeclareFontShape{OMX}{yhhex}{m}{n}{<->yhcmex10}{}
203 \DeclareSymbolFont{yhlargesymbols}{OMX}{yhhex}{m}{n}
204 \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.

```

205 \newcommand\ddfrac[2]{\frac{\,\displaystyle #1\,,}{\displaystyle #2\,,}}
206 \newcommand\numfrac[2]{\frac{#1}{#2\mathstrut}}
207 \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`.

```

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

```

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

```

209 \if@blacktriangleright
210   \def\labelitemi{\small $\blacktriangleright$}
211 \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.

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

`esercizi` 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` us to change the label style of all test environments, even in the “`shortlabels`”
`test-orizz` `enumitem` mode.
`\labeltest`

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

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

```
238 \newenvironment{test-orizz-newline}[1] []{%
239   \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).

```

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

```

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.

```

245 \def\@fattorevf{.75}
246 \def\@fattorevferror{%
247   \ClassError{verifica}{%
248     \protect\fattorevf\space e fattorevf devono avere come argomento
249     un\MessageBreak numero compreso tra 0 e 1}{%
250     Riprova scrivendo \protect\fattorevf{.85} o fattorevf=.85
251     o qualcosa di simile. Ciao ;) \MessageBreak}%
252 }
253 \newcommand{\fattorevf}[1]{%
254   \ifdim0pt<#1pt\relax
255     \ifdim1pt>#1pt\relax
256       \def\@fattorevf{#1}
257     \else
258       \@fattorevferror
259     \fi
260   \else
261     \@fattorevferror
262   \fi
263 }

```

\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).

```

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

```


\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.

```

276 \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 `\@fattere vf`). 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.

```

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

```

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

```

285 \enitkv@key{\fattere vf}{\fattere vf{#1}}
286 \newlist{test-vf}{enumerate}{1}
287 \setlist[test-vf]{label=\alph*}
288 \newcommand{\vfitem}[1]{\item\verofalso{#1}}
289 \newenvironment{test-verofalso}[1][\@fattere vf]{%
290   \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** `\dotword{word}` command inserts a dotted line that is twice the width of the word specified as an argument.

```

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

```

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

```

9.7 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).

```

303 \newcommand{\@punteggio}{Punti}
304 \newcommand{\punti}[2][\@nil]{%
305   \hfill\vphantom{,}\nolinebreak[1]\null~\hfill%
306   [\textsc{\@punteggio~#2}%
307     \def\tmp{#1}%
308     \ifx\tmp\@nnil
309       \relax
310     \else
311       ~\mbox{#1}%
312     \fi}}%
313 }

```

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

```

314 \newcommand{\punti@dx}[2][\@nil]{%
315   \hfill\vphantom{,}\nolinebreak[1]\null~\hfill%
316   [\textsc{#2~\@punteggio}%
317     \def\tmp{#1}%
318     \ifx\tmp\@nnil
319       \relax
320     \else
321       ~\mbox{#1}%
322     \fi}}%
323 }
324 \def\puntiadestra{\renewcommand{\punti}{\punti@dx}}

```

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

```

325 \def\puntiPt{\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.

```

326 \newcommand{\totpunti}[1][ /10 + 1 ]{%
327   \vfill
328   \begin{flushright}

```

```

329 {\renewcommand{\arraystretch}{1.5}
330 \begin{tabular}{|cc|}
331 \hline
332 & {\hspace{-7pt}\textbf{VOTO}} \\\
333 {\textbf{Totale punti} \fbox{\phantom{\LARGE MM}} $\bm{\#1=}$}%
334 & \hspace{-7pt}\fbox{\phantom{\Huge MM}} \\\
335 \hline
336 \end{tabular}}
337 \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.

```

338 \newcommand{\vartotpunti}[1][10 + 1]{%
339 \begin{flushright}
340 {\renewcommand{\arraystretch}{1.5}
341 \begin{tabular}{|cc|}
342 \hline
343 & {\hspace{-7pt}\textbf{VOTO}} \\\
344 {\textbf{Totale punti} \fbox{\phantom{\LARGE MM}} $\bm{\#1=}$}%
345 & \hspace{-7pt}\fbox{\phantom{\Huge MM}} \\\
346 \hline
347 \end{tabular}}
348 \end{flushright}}

```

10 Change History

v1.0		draws a dotted line for writing	
General: First public release 1	the candidate's name. 27
v1.1			
\lineanome: Added the		test-verofalso: Added the	
\lineanome command, which		test-verofalso environment	33

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.

Symbols		
\@classe 90,	\@classe 70, 71, 89, 90, 115, 116, 144, 145
92, 104, 116, 118,		\@completeleaders .. 66, 67, 83, 135, 162
136, 145, 147, 163	 293, 296, 298
\@lineanome	... 95,	\@data 68, 69, 84, 137, 164
97, 103, 121, 124,		\@disciplina 74, 75, 94, 120, 149
132, 150, 153, 161		\@disciplina 64, 65, 82, 139, 166
\@blacktrianglerightfalse	\@extrasisize true 9
..... 11	\@fattorevf	245, 256, 278
\@blacktrianglerighttrue	\@fattorevferror	.. 303, 306, 316, 325
..... 10 246, 258, 261	\@rigaleaders . 299, 302
\@boxcountlines	265, 267	\@intestright true .. 12
		\@risp 223, 227, 230

\@seconda .. 77, 88, 100	test-verofalso . 285	R
\@senzagrazi>true .. 14	\eps 181, 181	\R 181, 184
\@tabelladue ... 99, 113	esercizi (environ-	\raisebox 95
\@tabellauno 79, 87, 113	ment) 224	\Repeat ... 276, 282, 302
\@tempo	F	\restorephi 54, 58
72, 73, 107, 109,	\fattorevf 245, 285	\restoretheta ... 54, 59
126, 128, 155, 157	I	\riga 299
\@tempvarphi 54, 58	\if@blacktriangleright	T
\@tempvartheta .. 55, 59 1, 209	\tempo 62, 73
\@tipologia	\if@extrasize 4, 17, 219	\tempo@ad . 108, 111,
62, 63, 82, 139, 166	\if@intestrighe . 2, 169	113, 127, 130,
\@tratto .. 292, 294, 300	\if@intestsemplice 3, 172	142, 156, 159, 168
A	\if@sensagrazie 5, 40, 212	test (environment) .. 224
\abs 61, 61	\intestazione 169	test-orizz (environ-
\arc 189, 192	\intestazionedefault	ment) 224
\arc@arc 192, 193 76, 175	test-orizz-newline
\arc@char 191, 196	\intestazionerighe .	(environment) 238
B 114, 170	test-verofalso (envi-
\biglabelbox 212, 242, 243	\intestazioneemplice	ronment) ... 285
C 143, 173	\thelinenumber 271
\CalcNumber ... 264, 281	\istituto 62, 67	\themylines 274
\classe 62, 71	L	\tipologia 62, 63
D	\labelbox 212	\totpunti 326
\data 62, 69	\labeltest 224	\tsa 178, 179
\ddfrac 205	\lineanome 62, 75	\tso 178, 180
\Del 181, 183	M	U
\del 181, 182	\myx 281	\unitx 208, 208
\denfrac 205	N	V
\disciplina 62, 65	\N 181, 187	\vararc 201, 204
\dotrule 95, 122, 151, 291	\numfrac 205	\vartotpunti 338
\dotword 291	P	\ver@ptsize .. 6, 7, 8, 20
\dss 181, 188	\punti 303, 324	\verofalso 277, 288
E	\punti@dx 314, 324	\vf 240, 240, 282
environments:	\puntiadestra 314	\vfitem 288
esercizi 224	\punti@pt 325	\vskip 122, 151
test 224	Q	Z
test-orizz 224	\Q 181, 185	\Z 181, 186
test-orizz-newline		
..... 238		