

Il pacchetto verifiche *

v6.0

Francesco Raccanello
raccanello.francesco@gmail.com

17 settembre 2022

Abstract

The aim of this package is to manage the elements of a middle or high school test. It can compose the headings of the page, the exercises and their solutions. The exercises can show numerical ID, difficulty and score. The package manages some standard test typology: plain exercise, complete-the-text, true or false, open questions, closed questions, find-the-error.

Sommario

Il presente pacchetto si propone di gestire gli elementi che generalmente compongono una prova scritta per le scuole secondarie di primo e secondo grado: l'intestazione della pagina, gli esercizi della verifica e le loro soluzioni. Gli esercizi possono presentare una numerazione, un punteggio e un grado di difficoltà. Alcuni comandi permettono di generare alcune tipologie di test: esercizi semplici, esercizi di completamento, vero o falso, domande aperte e chiuse e trova l'errore. Ciascun esercizio può presentare la soluzione corretta che viene riportata nel documento in base ad un'apposita opzione del pacchetto.

*Questo documento corrisponde al pacchetto verifiche v6.0, in data 2022/09/17.

Indice

1	Introduzione	3
2	Come usare il pacchetto <code>verifiche</code> e le sue opzioni	3
3	Intestazione	4
4	Comandi ed ambienti per le verifiche	5
5	Personalizzazione	12
6	Test multipli	15
7	Implementation	17
7.1	Require packages	17
7.2	Packages and library options	17
7.3	<code>newif</code>	17
7.4	Package options declaration	18
7.5	Counters	18
7.6	Length	18
7.7	Labels	19
7.8	Macro and environment	19

1 Introduzione

Il pacchetto `verifiche` nasce dalla necessità di creare verifiche cartacee per la scuola secondaria di primo e secondo grado. L'intento è di ottenere un supporto che controlli la parte di test, nonché la gestione delle soluzioni all'interno dello stesso documento.

Riguardo la parte di esercizi, il presente pacchetto si occupa della suddivisione delle domande (numerata o meno) e della gestione del loro punteggio inserendolo a mano, oppure calcolandolo come totale di punteggi parziali (si veda pagina 6). Sono presenti comandi o ambienti ad hoc per realizzare quesiti di vero/falso, domande aperte o chiuse, esercizi di "caccia all'errore" e la gestione di testi e immagini affiancate.

Per quanto riguarda le soluzioni, ogni elemento di test presenta la possibilità di mostrare la soluzione corretta, qualora venga inserita una specifica opzione del pacchetto che attivi la così detta in "modalità `solutions`" indicata nel seguito anche con "modalità soluzioni".

Infine, il codice carica alcuni pacchetti necessari per la gestione dei comandi implementati, ma anche per rendere fruibili da subito comandi utili soprattutto per la gestione della matematica, le unità di misura, e la parte grafica. Nel seguito si riportano i pacchetti principali:

- `enumitem` per la gestione avanzata degli elenchi puntati e numerati (sono attivate le opzioni `shortlabels` e `inline`)
- `amsmath`, `amssymb` per la gestione della matematica;
- `siunitx` per la gestione delle unità di misura;
- `graphicx`, `xcolor` e `tikz` per la gestione grafica;

2 Come usare il pacchetto `verifiche` e le sue opzioni

Per usare il pacchetto `verifiche` è sufficiente copiare il file `verifiche.sty` nella cartella di lavoro o installarlo secondo le procedure relative al proprio sistema operativo.

Una volta installato è sufficiente caricare il pacchetto attraverso il comando:

```
\usepackage[<options>]{verifiche}
```

Il pacchetto `verifiche` prevede alcune opzioni per modificare globalmente il proprio comportamento. Queste gestiscono l'aspetto dell'intestazione, la numerazione globale dei quesiti nonché l'attivazione della "modalità soluzioni" per poter visualizzare, nel documento, le soluzioni e variarne il colore del font.

<i>⟨options⟩</i>	Descrizione
<code>noinstitute</code>	non stampa il nome dell'istituto.
<code>nocandidatename</code>	non stampa la sezione per il nome del candidato
<code>noinstruction</code>	non stampa le istruzioni della prova
<code>noduration</code>	non stampa la durata della prova
<code>noasyear</code>	non stampa l'anno scolastico
<code>notesttype</code>	non stampa il tipo di prova
<code>nonumbered</code>	quesiti non numerati
<code>noexercisept</code>	non stampa il punteggio degli esercizi, anche se dichiarati
<code>nosubtitle</code>	non stampa il sottotitolo
<code>solutions</code>	stampa le soluzioni degli esercizi (si veda pag. 8)
<code>red</code>	il colore del font delle soluzioni è rosso
<code>blue</code>	il colore del font delle soluzioni è blu
<code>color=⟨color⟩</code>	il colore del font delle soluzioni è <i>⟨color⟩</i>

Nel seguito si indicherà con "modalità test", il pacchetto caricato senza l'opzione `solutions`, altrimenti questa verrà detta "modalità soluzione" o "modalità solutions".

3 Intestazione

Comandi

Il pacchetto `verifiche` fornisce alcuni comandi per realizzare una semplice intestazione del documento che prevede il nome dell'istituto, l'anno scolastico il tipo di prova, le istruzioni e la durata della prova. Tutte queste voci vanno dichiarate prima del comando `\printhead` che si occupa della produzione del risultato finale. Le *opzioni* del pacchetto permettono la soppressione della visualizzazione di alcune di queste voci.

`\institute` Il nome dell'istituto può essere indicato tramite il comando:

```
\institute{⟨nome dell'istituto⟩}
```

il cui argomento verrà stampato al centro dell'intestazione della pagina con un font largo, senza grazie e di colore grigio scuro. Il font può essere modificato ridefinendo il comando `\institutefont`.

`\asyear` L'anno scolastico viene indicato tramite l'apposito comando:

```
\asyear[⟨preambolo⟩]{⟨anno scolastico⟩}
```

il cui argomento viene riportato, al centro della pagina, antepoendo il *⟨preambolo⟩* (che di default riporta la dicitura "Anno scolastico"). Il font ed il posizionamento del testo possono essere cambiati ridefinendo il comando `\asyearfont`.

`\testtype` La tipologia di prova può essere indicata tramite il comando:

```
\testtype{⟨tipo di compito⟩}
```

il cui argomento viene riportato, per default, al centro della pagina. La modifica del font e del posizionamento può essere operato ridefinendo il comando `\testtypefont`.

`\instruction` Le istruzioni della prova possono essere racchiuse all'interno del comando:

`\instruction{<istruzioni>}`

che provvede a racchiudere tra due delimitatori (di default una riga orizzontale) il testo delle istruzioni. Ridefinendo il comando `\instructiofont` è possibile cambiare il font delle istruzioni.

`\duration` La durata della prova può venir indicata tramite il comando:

`\duration[<preambolo>]{<durata>}`

che viene riportata sotto le istruzioni, al centro, antepoendo all'argomento il `<preambolo>` (che di default riporta la dicitura "Tempo della prova:"). L'aspetto di questo comando può essere modificato ridefinendo il comando `\durationfont`.

`\subtitle` Il comando `\subtitle` introduce un sottotitolo al di sotto di quanto visualizzato dal comando `\testtype`. L'aspetto ed il posizionamento di questo testo possono essere modificati ridefinendo il comando `\subtitlefont`.

`\subtitle{<testo del sottotitolo>}`

`\printheadng` L'intestazione, ovvero l'output di tutti i comandi appena definiti, viene riportata nel documento solo in presenza del comando

`\printheadng`

Riassumendo, il pacchetto fornisce i seguenti comandi per gestire e la stampa dell'intestazione del documento:

Comando	Font	Descrizione
<code>\institute</code>	<code>\institutefont</code>	Definisce il nome dell'istituto.
<code>\asyear</code>	<code>\asyearfont</code>	Definisce l'anno scolastico.
<code>\testtype</code>	<code>\testtypefont</code>	Definisce il tipo di prova.
<code>\instruction</code>	<code>\instructiofont</code>	Definisce le istruzioni delle prove.
<code>\duration</code>	<code>\durationfont</code>	Definisce la durata della prova.
<code>\subtitle</code>	<code>\subtitlefont</code>	Definisce il sottotitolo.
<code>\printheadng</code>		Stampa l'intestazione.

Stili di pagina

Il pacchetto definisce due stili di pagina *ad hoc*: `mainverifiche` e `verifiche`. Il primo è evocato quando si definisce il comando `\institute` e stampa sulla prima pagina l'argomento di questo comando. Il secondo, invece, può essere utilizzato dall'utente tramite il comando:

`\pagestyle{verifiche}`

e riproduce, nelle pagine dispari diversa dalla prima, l'intestazione per l'identificazione del candidato: il nome, la classe e la data.

4 Comandi ed ambienti per le verifiche

Nel seguito verranno presentati i comandi che permettono di strutturare un test. Il pacchetto fornisce un ambiente di esercizio in cui racchiudere le domande. Vi è la possibilità di predisporre domande chiuse a scelta multipla, prove di vero-falso, "trova l'errore" e domande a risposta aperta (fornendo una base su cui scrivere

come un foglio a quadretti o a righe). Ogni comando prevede la possibilità di fornire le soluzioni che vengono visualizzato quando il pacchetto è caricato attivando l'opzione `solutions`.

`esercizio` (*env.*) Gli esercizi od i quesiti del test vengono gestiti dall'ambiente `esercizio`.

```
\begin{esercizio}[label=<label>,diff=<number>,pt=<number>,partialpt]
...
\end{esercizio}
```

Senza specificare alcun argomento opzionale, viene stampata la dicitura "Quesito" seguita da un indice numerico crescente ed il testo contenuto nel corpo dell'ambiente. Le modifiche dell'aspetto e del comportamento sono previste tramite delle opzioni espresse nel formato `key=val`, eventualmente separate da virgola se presenti in contemporanea:

<code>label=<label></code>	definisce il titolo del quesito
<code>diff=<number></code>	definisce la difficoltà sotto forma di stelle
<code>pt=<number></code>	definisce il punteggio della prova
<code>partialpt</code>	somma in automatico i punteggi parziali

Nel seguito si mostrano alcuni dei possibili risultati visivi.

<p>Esempio di codice</p> <pre>\begin{esercizio} Testo dell'esercizio che voglio somministrare \end{esercizio}</pre>	<p>Esempio visivo</p> <p>Quesito 1 Testo dell'esercizio che voglio somministrare</p>
<p>Esempio di codice</p> <pre>\begin{esercizio}[label=Esercizio] Testo dell'esercizio che voglio somministrare \end{esercizio}</pre>	<p>Esempio visivo</p> <p>Esercizio 2 Testo dell'esercizio che voglio somministrare</p>
<p>Esempio di codice</p> <pre>\begin{esercizio}[diff=2] Testo dell'esercizio che voglio somministrare \end{esercizio}</pre>	<p>Esempio visivo</p> <p>★★ Quesito 3 Testo dell'esercizio che voglio somministrare</p>

Esempio di codice	Esempio visivo
<pre>\begin{esercizio}[pt=2] Testo dell'esercizio che voglio somministrare \end{esercizio}</pre>	<p style="text-align: right;">___/ 2 pt Quesito 4 Testo dell'esercizio che voglio somministrare</p>
Esempio di codice	Esempio visivo
<pre>\begin{esercizio}[pt=2, diff=2] Testo dell'esercizio che voglio somministrare \end{esercizio}</pre>	<p style="text-align: right;">★★ Quesito 5 ___/ 2 pt Testo dell'esercizio che voglio somministrare</p>

È possibile fare dei riferimenti incrociati usando gli usuali comandi presenti in \LaTeX . L'etichetta viene dichiarata tramite l'usuale comando `\label={\langle label \rangle}` introdotto dopo l'apertura dell'ambiente `esercizio`. I riferimenti incrociati possono essere richiamati tramite l'usuale comando `\ref{\langle label \rangle}`, che produce il numero dell'esercizio e `\pageref{\langle label \rangle}` che stampa la pagina del riferimento.

`\partialpt` Il comando `\partialpt`, da non confondere con l'opzione omonima `partialpt` dell'ambiente `esercizio`, permette di riportare il punteggio parziale per le domande di un esercizio. Questo viene stampato a margine e racchiuso tra parentesi tonde.

Esempio di codice	Esempio visivo
<pre>\begin{esercizio} \begin{enumerate} \item\partialpt{1} Domanda 1 \item\partialpt{3} Domanda 2 \end{enumerate} \end{esercizio}</pre>	<p style="text-align: center;">Quesito 6</p> <p>(1 pt) 1. Domanda 1</p> <p>(3 pt) 2. Domanda 2</p>

Quando usato negli ambienti `enumerate` o `itemize`, il comando introduce uno spazio spurio che può essere rimosso posponendo il comando `\noindent`. Quando viene esplicitata l'opzione `partialpt` nella definizione dell'ambiente `esercizio`, il pacchetto si occupa della somma complessiva dei singoli punteggi parziali riportandola come punteggio totale del quesito. In questo caso il documento dev'essere compilato *due volte* per ottenere correttamente il risultato finale.

Esempio di codice	Esempio visivo
<pre>\begin{esercizio}[partialpt] \begin{enumerate} \item\partialpt{1} Domanda 1 \item\partialpt{3} Domanda 2 \end{enumerate} \end{esercizio}</pre>	<p>___/4 pt Quesito 7</p> <p>(1 pt) 1. Domanda 1</p> <p>(3 pt) 2. Domanda 2</p>

Il punteggio dei quesiti può essere richiamato in qualunque punto del documento attraverso il comando

$$\backslash\text{ref}\{\text{ptes}@<n.es>\}$$

dove $\langle n.es \rangle$ indica il numero dell'esercizio riportato nell'intestazione dello stesso.

esercizio* (*env.*) L'ambiente `esercizio` produce sempre quesiti numerati. Se si desidera indicare test non numerati è possibile farlo tramite l'ambiente `esercizio*` che prevede gli stessi argomenti opzionali definiti per l'ambiente precedente.

```
\begin{esercizio*}[label=<label>,diff=<number>,pt=<number>,partialpt]
...
\end{esercizio*}
```

Esempio di codice	Esempio visivo
<pre>\begin{esercizio*}[pt=2] Testo dell'esercizio che voglio somministrare \end{esercizio*}</pre>	<p>___/ 2 pt Quesito</p> <p>Testo dell'esercizio che voglio somministrare</p>

soluzione (*env.*) L'ambiente `soluzione` racchiude al suo interno tutto il materiale utile ad essere visualizzato solo in presenza dell'opzione `solutions` del pacchetto `verifiche`. Quando il pacchetto è in modalità "test" il contenuto dell'ambiente `soluzione` viene ignorato. L'etichetta che introduce l'ambiente può essere modificata localmente tramite l'opzione `label`.

```
\begin{soluzione}[label=<label>]
...
\end{soluzione}
```

Esempio di codice	Esempio visivo
<pre>\begin{soluzione} Testo della soluzione dell'esercizio somministrato \end{soluzione}</pre>	<p>Soluzione</p> <p><i>Testo della soluzione dell'esercizio somministrato</i></p>

`\inlinesol` Il pacchetto presenta il comando `\inlinesol` per produrre soluzioni in linea, ov-

vero testo che in modalità "test" viene ignorato, ma che invece viene visualizzato nella modalità "soluzioni" (opzione `solutions` del pacchetto).

```
\inlinesol{<testo>}
```

Esempio di codice	Esempio visivo
Questa e' una <code>\inlinesol{soluzione}</code> in linea	Questa e' una in linea. Questa e' una <i>soluzione</i> in linea

`\completetext` Gli esercizi di completamento del testo possono venir realizzati tramite il comando:

```
\completetext[<dim>]{<testo>}
```

che produce uno spazio vuoto largo $\langle dim \rangle$ (che di default è di 4,5 cm) e che in modalità "soluzioni" riproduce il $\langle testo \rangle$.

Esempio di codice	Esempio visivo
La <code>\completetext[2cm]{soluzione}</code>	La _____ La <i>soluzione</i>

`\truefalse` I quesiti con domande vero-falso possono essere realizzati tramite il comando

```
\truefalse{<V/F>}
```

Il comando produce due riquadri quadrati allineati con le lettere V e F; l'argomento obbligatorio richiede di indicare la lettera V o F al fine di poter realizzare un segno di spunta (✓) sulla casella giusta nella modalità `solutions` del pacchetto. Tale

simbolo può essere modificato ridefinendo il comando `\checkmaker`.

```
\renewcommand{\checkmaker}{<...>}
```

Esempio di codice	Esempio visivo
Vero o Falso? <code>\truefalse{V}</code>	Vero o Falso? <input type="checkbox"/> V <input type="checkbox"/> F Vero o Falso? <input checked="" type="checkbox"/> V <input type="checkbox"/> F

`closedquestion` (*env.*) Le domande a scelta multipla possono essere realizzate tramite l'ambiente `closedquestion`. Questo comando genera un elenco verticale di elementi cia-

scuno introdotto dal comando `\item` che realizza un quadrato davanti ad ogni voce.

```
\begin{closedquestion}[...]
\item ...
\end{closedquestion}
```

`\checked` La soluzione (o le soluzioni) possono essere segnate tramite il comando `\checked` posto come argomento opzionale del comando `\item` e che, nella modalità `solutions`, produce un segno di spunta (✓) sulla casella selezionata. Anche in questo caso tale simbolo può essere modificato ridefinendo il comando `\checkmaker`. Il pacchetto ammette degli argomenti opzionali che possono essere tutti quelli descritti nel pacchetto `enumitem`.

```
\renewcommand{\checkmaker}{<...>}
```

Esempio di codice	Esempio visivo
<pre>\begin{closedquestion} \item Primo \item[\checked] Secondo \item Terzo \item Quarto \end{closedquestion}</pre>	<input type="checkbox"/> Primo <input checked="" type="checkbox"/> Secondo <input type="checkbox"/> Terzo <input type="checkbox"/> Quarto

`closedquestion` (*env.*) L'ambiente `closedquestion*` è analogo al precedente, fatto salvo che è adibito a produrre elenchi in linea; gli elementi vengono spaziati orizzontalmente in modo da riempire tutto lo spazio disponibile. Anche in questo caso i comandi del pacchetto `enumitem` possono essere usati tra le opzioni dell'ambiente.

```
\begin{closedquestion*}[...]
\item ...
\end{closedquestion*}
```

`\openquestion` Il comando produce una base di supporto per la scrittura, ovvero delle linee o dei quadretti. La sintassi del comando è la seguente:

```
\openquestion[type=<lines/squares>, width=<dim>, height=<dim>,
spacedim=<dim>, linecolor=<color>]{<soluzione>}
```

La il testo dell'argomento obbligatorio (*<soluzione>*) viene visualizzato solo in modalità `solutions`. Il comando gestisce alcune opzioni:

<code>type</code>	accetta due valori: <code>lines</code> e <code>squares</code> (default: <code>squares</code>)
<code>width</code>	gestisce la dimensione orizzontale del campo di testo (default: 12 cm)
<code>height</code>	gestisce la dimensione verticale del campo di testo (default: 20 cm)
<code>spacedim</code>	gestisce la dimensione dei quadretti o delle linee (default: 0.5 cm)
<code>linecolor</code>	colore delle linee guida (default: gray)

Esempio di codice	Esempio visivo
<pre>\openquestion{ Prova del testo della soluzione della domanda aperta }</pre>	

Esempio di codice	Esempio visivo
<pre>\openquestion[type=lines]{ Prova del testo della soluzione della domanda aperta }</pre>	

`\finderror` Gli esercizi della tipologia “trova l'errore” possono essere composti tramite il comando

```
\finderror{<testo sbagliato>}{<testo giusto>}
```

Il comando visualizza la stringa `<testo sbagliato>` nella modalità "test", mentre mostra una cancellatura ed il testo corretto nella modalità `solutions`.

Esempio di codice	Esempio visivo
<pre>\finderror{sbagliato}{corretto}</pre>	<pre>corretto sbagliato</pre>

`\textandimage` L'impaginazione del testo con affiancato un'immagine viene agevolato dal comando `\textandimage` che ha la seguente sintassi:

```
\textandimage[<left width>][<right width>]{<sx material>}{<dx material>}
```

Il `<rx material>` viene affiancato all'`<dx material>`. La larghezza della colonna sinistra (risp. destra) è controllata tramite l'argomento opzionale `<left width>` (risp. `<right width>`).

Esempio di codice	Esempio visivo
<pre>\textandimage[2cm][1cm]{Testo}{ \tikz{\draw(0,0) circle (.4cm)}}</pre>	<pre>Testo </pre>

`\makecolumn` Il pacchetto `verifiche` definisce anche un proprio comando per produrre del testo in colonna tramite il comando `\makecolumn` che ha la seguente sintassi:

```
\makecolumn{<width>}{<text>}
```

La larghezza è controllata tramite l'argomento opzionale `<width>`, mentre il testo da mettere in colonne può risiedere, anche con paragrafi, entro il campo `<text>`. La spaziatura tra le colonne può essere modificata ridefinendo la lunghezza `\columnseparation` (di default viene inizializzata col valore `\hfill`).

```
\setlength{\columnseparation}{<dim>}
```

Esempio di codice	Esempio visivo
<pre>\makecolumn{.3\textwidth}{Colonna 1} \makecolumn{.3\textwidth}{Colonna 2}</pre>	<pre>Colonna 1 Colonna 2</pre>

5 Personalizzazione

Il pacchetto prevede un certo numero di comandi per gestire l'aspetto del documento finale.

Font

Per l'intestazione tutti i font sono definiti secondo lo schema:

<comando intestazione>font

ed oltre a gestire l'aspetto del carattere, alcuni di questi, specificano la posizione occupata del testo ed eventuali prefissi.

`\headerfont` Il font delle intestazioni quando viene usato lo stile di pagina `verifiche`, può essere modificato ridefinendo il comando `\headerfont`:

```
\renewcommand{\headerfont}[1]{\dots}
```

`\institutefont` Il font dell'istituto, generato dal comando `\institute`, è gestito dal comando `\institutefont`. Per variare le caratteristiche del font è necessario ridefinire il comando:

```
\renewcommand{\institutefont}[1]{\dots}
```

`\asyearfont` Il font dell'anno scolastico, generato dal comando `\asyear`, è gestito dal comando `\asyearfont`. Per variare le caratteristiche del font è necessario ridefinire il comando:

```
\renewcommand{\asyearfont}[1]{\dots}
```

`\testtypefont` Il font tipo di test, generato dal comando `\testtype`, è gestito dal comando `\testtypefont`. Per variare le caratteristiche del font è necessario ridefinire il comando:

```
\renewcommand{\testtypefont}[1]{\dots}
```

`\instrunctionfont` Il font delle istruzioni, generate dal comando `\instrunction`, è gestito dal comando `\instrunctionfont`. Per variare le caratteristiche del font è necessario ridefinire il comando:

```
\renewcommand{\instrunctionfont}[1]{\dots}
```

`\solutionfont` Le soluzioni degli esercizi vengono stampate, quando viene esplicitata l'opzione `solutions` quando viene caricato il pacchetto, con un font più piccolo e in corsivo. Per modificare il font delle soluzioni è sufficiente ridefinire il comando `\solutionfont`:

```
\renewcommand{\solutionfont}{\dots}
```

Intestazione

`\instructiondelimiter` Le istruzioni e la durata della prova sono racchiuse tra due delimitatori che normalmente sono due righe orizzontali. Per cambiare, o rimuovere, questi delimitatori è sufficiente ridefinire il seguente comando:

```
\renewcommand{\instructiondelimiter}{\dots}
```

`\headercandidatenamelabel` Quando viene composta l'intestazione principale nella prima pagina, compare una riga in cui il candidato può scrivere il proprio nome e cognome, la data e la classe. L'etichetta della prima voce è determinata dal comando `\headercandidatenamelabel`, che può essere opportunamente cambiata tramite il comando:

```
\renewcommand{\headercandidatenamelabel}{\langle...\rangle}
```

`\headerclasslabel` Allo stesso modo è possibile modificare la descrizione della classe tramite la redefinizione del comando `\headerclasslabel` e della data tramite comando `\headerdatelabel`. Tutte queste tre voci prevedono, per ciascuna, una riga vuota ove scrivere l'informazione richiesta. Le lunghezze delle tre righe possono essere modificate impostando le lunghezze desiderate

`\candidatenerulerlength`
`\classrulerlength`
`\daterulerlength`

```
\setlength{\langle...rulerlength\rangle}{\langle dim\rangle}
```

Attraverso questi comandi è possibile anche rimuovere una (o più) delle tre voci. Se ad esempio non si vuole visualizzare la classe d'appartenenza del candidato, sarà sufficiente ridefinire gli opportuni comandi nel seguente modo: `\renewcommand{\headerclasslabel}{}`, così da non visualizzare alcuna scritta, e `\setlength{\classrulerlength}{0pt}` per rimuovere la riga associata.

Ambiente esercizio

`\exerciselabel` Gli esercizi vengono inseriti antepoendo al loro testo la dicitura "Quesito" seguito (eventualmente) da un numero. Per cambiare questa dicitura, a livello globale, ovvero per ogni esercizio, è sufficiente ridefinire il comando `\exerciselabel`.

```
\renewcommand{\exerciselabel}{\langle Preambolo dell'esercizio\rangle}
```

L'ambiente `esercizio` permette di riportare anche il punteggio complessivo e parziale per le sotto-domande. Di default il punteggio totale del quesito viene riportato antepoendo un tratto orizzontale seguito da uno slash il valore del punteggio ed un'etichetta indicante la sigla *pt*. Per quanto riguarda il punteggio parziale, questo viene riportato racchiuso tra parentesi tonde. L'aspetto complessivo del punteggio, quindi, può essere così schematizzato:

prefisso delimitatore punteggio label delimitatore

`\ptprefix` Il comando `\ptprefix` permette di modificare il `prefisso` del punteggio. Di default traccia una linea, la cui lunghezza può essere variata modificando la lunghezza `ptrulerlength`, ed uno slash. Se si vuole modificarne l'aspetto è sufficiente evocare il comando.

```
\ptprefix{\langle...\rangle}
```

`\partialptprefix` Allo stesso modo il comando `\partialptprefix` definisce un `prefisso` per il punteggio parziale.

`\ptdelimiters` I punteggi sono racchiusi tra due `delimitatori`. Il punteggio totale non presenta alcun simbolo, mentre quello parziale è racchiuso tra parentesi tonde. È possibile definire dei nuovi `delimitatori` utilizzando i comandi `\partialptdelimiters` e `\ptdelimiters` i quali argomenti contengono due caratteri che fungono da delimitatore dentro e sinistro.

```
\ptdelimiters{\langle coppia di caratteri\rangle}  

\partialptdelimiters{\langle coppia di caratteri\rangle}
```

`\ptlabel` Il punteggio viene riportato indicando anche una `label`. Questa di default è "pt"
`\partialptlabel` sia per il punteggio parziale che per quello totale. È possibile definire un nome alternativo usando il comando `\ptlabel` per il punteggio totale o `\partialptlabel` per quello parziale. I comandi prevedono un argomento obbligatorio che è valido per tutti i punteggi, ma anche uno facoltativo che viene usato quanto il valore del punteggio è pari a 1 e che necessita quindi di un sostantivo singolare al posto di quello plurale (i.e. "punto" al posto di "punti").

```
\ptlabel[{sing.}]{{plur.}}  

\partialptlabel[{sing.}]{{plur.}}
```

`\diffsymb` L'ambiente esercizi prevede di poter descrivere la difficoltà tramite un simbolo grafico. Di default questo è una stella (★), tuttavia è possibile cambiare il tipo di simbolo ridefinendo il comando `\diffsymb`:

```
\renewcommand{\diffsymb}{{codice del simbolo}}
```

Ambiente `closedquestion`

`\closedquestionitem` Il simbolo delle domande chiuse è controllato dal comando `\closedquestionitem` che di default è un quadrato (□). Per modificare questo simbolo è sufficiente ridefinire il comando:

```
\renewcommand{\closedquestionitem}{{codice del simbolo}}
```

Il segno di spunta che viene stampato nella modalità soluzione, può essere cambiato ridefinendo il comando `\checkmaker`

```
\renewcommand{\checkmaker}{{...}}
```

Ambiente `soluzione`

Per quel che riguarda l'ambiente `soluzione` è possibile cambiare il nome dell'ambiente ridefinendo il comando `\solutionlabel`.

```
\renewcommand{\solutionlabel}{{Preambolo della soluzione}}
```

6 Test multipli

Uno degli aspetti peculiari della produzione di test tramite software tipografici come \LaTeX è quello di poter usufruire della sua versatilità per produrre un numero definito a priori dall'utente di test diversi a partire da uno stesso schema di base. Le modalità e le strategie per fare questo sono molteplici. Uno di questi metodi è utilizzare il pacchetto¹ `datatool` che permette di leggere un database (solitamente salvato come file `csv`) al cui interno sono contenute le domande, i dati ed eventualmente le soluzioni degli esercizi. Il pacchetto, poi, mette a disposizione il comando `\DTLforeach` che, interrogando tutto il database precedentemente costruito, permette di popolare un unico documento in cui si variano taluni elementi al fine di ottenere prove didattiche differenti. Una soluzione di questo tipo necessita di scrivere almeno due file. Il primo, con estensione `tex`, è la struttura della prova di verifica al cui interno presenta tutti i comandi per utilizzare i dati contenuti nel secondo file. Quest'ultimo è un file di tipo `csv` al cui interno, organizzate per colonne, si trovano tutte le domande che si vogliono personalizzare ed ogni riga corrisponde ad un differente compito. Il pacchetto permette anche di generare un database direttamente interno a \LaTeX , senza quindi dover creare un file a parte, tramite i comandi:

```
\DTLnewdb{nome del database}
```

per creare un database chiamato `nome del database`; per creare una nuova riga, corrispondente ad una prova diversa, si usa il comando

```
\DTLnewrow{nome del database}
```

e ciascuna entrata di questa riga viene popolata introducendo un'entrata tramite il comando:

```
\DTLnewdbentry{nome del database}{nome colonna}{dato}
```

Come nel caso precedente il database può essere utilizzato tramite il comando²

```
\DTLforeach*{nome del database}{comando=nome colonna}{dato}{testo}
```

dichiarato subito dopo `\begin{document}` ed il `testo` non è altro che l'intero corpo del documento.

L'ambiente `multitest`

Il pacchetto `verifiche` mette a disposizione una soluzione interna per generare un numero definito dall'utente di prove personalizzate e che si basa sugli strumenti `multitest` (*env.*) messi a disposizione da `pgf` e `tikz`. L'ambiente `multitest` racchiude all'interno del suo corpo lo schema della prova che si vuole somministrare ed eventualmente i comandi che permettono di personalizzare la prova.

```
\begin{multitest}[n]  
...  
\end{multitest}
```

¹<https://ctan.org/pkg/datatool>

²Si rimanda alla documentazione ufficiale per ulteriori approfondimenti. Si faccia attenzione che l'ultima voce del secondo argomento del comando, ovvero `comando=nome colonna` non deve contenere spazi, si consiglia quindi di concludere la riga con il carattere `%`

L'ambiente accetta un argomento opzionale `n` che è il numero di prove che si vogliono realizzare. Normalmente l'ambiente viene aperto dopo `\begin{document}` e chiuso prima di `\end{document}`.

La personalizzazione dei dati del problema può avvenire tramite i comandi di gestione dei numeri casuali e delle liste casuali messi a disposizione dal pacchetto `pgf` che vengono di seguito riassunti lasciando il loro approfondimento alla documentazione ufficiale³.

```
\pgfmathrnd
\pgfmathrand
\pgfmathrandom{x,y}
\pgfmathgeneratepseudorandomnumber
\pgfmathrandominteger{macro}{min}{max}
\pgfmathdeclarerandomlist{nome lista}{ {item 1} {item 2} ...}
\pgfmathrandomitem{macro}{nome lista}
\pgfmathsetseed{numero intero}
```

Il comando inizializza il seme dei numeri casuali col numero del compito prodotto, in modo tale da produrre gli stessi compiti casuali ad ogni compilazione.

`multitestcounter` Il contatore `multitestcounter` permette di identificare il numero della traccia generata.

³<https://www.ctan.org/pkg/pgf>

7 Implementation

7.1 Require packages

The implementation of the `verifiche` package starts with the required packages:

```
1 \RequirePackage{xparse}
2 \RequirePackage{xkeyval}
3 \RequirePackage{amsmath}
4 \RequirePackage{amssymb}
5 \RequirePackage{pgffor}
6 \RequirePackage[shortlabels, inline]{enumitem}
7 \RequirePackage{siunitx}
8 \RequirePackage{graphicx}
9 \RequirePackage{xcolor}
10 \RequirePackage{tikz}
11 \RequirePackage{booktabs}
12 \RequirePackage[normalem]{ulem}
```

7.2 Packages and library options

The package loads `tikz` and some useful library. The `siunitx` package is tuned in order to be used for the Italian math document.

```
13 \usetikzlibrary{calc,shapes.geometric,patterns,positioning,angles,quotes}
14 \sisetup{
15   output-decimal-marker = {,},
16   exponent-product = \cdot,
17   per-mode=symbol-or-fraction,
18   separate-uncertainty
19 }
```

7.3 `newif`

Some conditionals are defined and initiated:

```
20 \newif\ifsol
21 \newif\ifinstitute\institutetrue
22 \newif\ifinstruction\instructiontrue
23 \newif\ifduration\durationtrue
24 \newif\ifasyear\asyeartrue
25 \newif\ifsolutionscolor\solutionscolorfalse
26 \newif\ifcandidatename\candidatename>true
27 \newif\iftesttype\testtypetrue
28 \newif\ifsubtitle\subtitltrue
29 \newif\ifshowinstructiondelimiter\showinstructiondelimiterfalse
30 \newif\ifprintedheading\printedheadingfalse
31 \newif\ifexercisesnumbered\exercisesnumberedtrue
32 \newif\ifshowmarginexercise\showmarginexercisefalse
33 \newif\ifshowexercisept\showexercisepttrue
34 \newif\ifopenquestionlines\openquestionlinesfalse
35 \newif\ifopenquestionsquared\openquestionsquaredfalse
```

7.4 Package options declaration

The package provides some options to control the aspect of the document and the mode of the document ("test" mode or "solutions" mode).

```
36 \DeclareOptionX{solutions}[]{\soltrue}
37 \DeclareOptionX{noinstitute}[]{\institutefalse}
38 \DeclareOptionX{noinstruction}[]{\instructionfalse}
39 \DeclareOptionX{noduration}[]{\durationfalse}
40 \DeclareOptionX{noasyear}[]{\asyearfalse}
41 \DeclareOptionX{nocandidatename}[]{\candidatenamefalse}
42 \DeclareOptionX{noteststtype}[]{\teststtypefalse}
43 \DeclareOptionX{nosubtitle}[]{\subtitlfalse}
44 \DeclareOptionX{nonumbered}[]{\exercisenumbefalse}
45 \DeclareOptionX{noexercisep}[]{\showexercisepfalse}
46 \DeclareOptionX{color}[red]{\solutionscolortrue\def\solutionscolor{#1}}
47 \DeclareOptionX{red}[]{\ExecuteOptionsX{color=red}}
48 \DeclareOptionX{blue}[]{\ExecuteOptionsX{color=blue}}
49 \DeclareOptionX*{\PackageWarning{verifich}{Unknown ‘\CurrentOption’}}
50 \ProcessOptionsX
```

7.5 Counters

The package use some predefined counters: the number of the exercise is controlled by `exercisenumbe`, while the partial points of an exercise are summed into `partialpoints` (and it's reseted every new exercise) to produce the sum of the partial points for a given exercise.

```
51 \newcounter{exercisenumbe}
52 \newcounter{partialpoints}[exercisenumbe]
```

7.6 Length

In order to fine-tuning the dimension of the rulers in the headers and to leave the possibility to the user to redefine their dimensions, some length are introduced.

```
53 \newlength{\candidatename rulerlength}
54 \newlength{\class rulerlength}
55 \newlength{\date rulerlength}
56 \newlength{\p rulerlength}
57 \newlength{\columnseparation}
58 \newlength{\verifich@remainingvspace}
59 \newlength{\verifich@remaininghspace}
60 \newlength{\verifich@truefalsesquares}
61 \newlength{\spacedim}
62 \setlength{\candidatename rulerlength}{.35\textwidth}
63 \setlength{\class rulerlength}{.1\textwidth}
64 \setlength{\date rulerlength}{.15\textwidth}
65 \setlength{\p rulerlength}{1cm}
66 \setlength{\columnseparation}{\fill}
67 \setlength{\verifich@truefalsesquares}{.7\baselineskip}
```

7.7 Labels

Some labels are define as a variable in order to leave the user the possibility of customization the predefine texts. If the package `babel` or `polyglossia` is loaded with the italian language specify, than the predefine labels are redefine:

```
68 \newcommand{\@verifichedurationlabeldefault}{Time:}%
69 \newcommand{\@verifichesyyearlabeldefault}{School year}%
70 \newcommand{\@verifichheadercandidatename}{Name and Surname}%
71 \newcommand{\@verifichheaderclasslabel}{Class}%
72 \newcommand{\@verifichheaderdatelabel}{Date}%
73 \newcommand{\@verifichexerciselabel}{Exercise}%
74 \newcommand{\@verifichsolutionlabel}{Solution}%
75 \newcommand{\@verifichTletter}{T}%
76 \newcommand{\@verifichFletter}{F}%
77 %
78 \newcommand{\italiandictionary}{%
79   \renewcommand{\@verifichedurationlabeldefault}{Tempo della prova:}%
80   \renewcommand{\@verifichesyyearlabeldefault}{Anno scolastico}%
81   \renewcommand{\@verifichheadercandidatename}{Nome e Cognome}%
82   \renewcommand{\@verifichheaderclasslabel}{Classe}%
83   \renewcommand{\@verifichheaderdatelabel}{Data}%
84   \renewcommand{\@verifichexerciselabel}{Quesito}%
85   \renewcommand{\@verifichsolutionlabel}{Soluzione}%
86   \renewcommand{\@verifichTletter}{V}%
87   \renewcommand{\@verifichFletter}{F}%
88 }%
89 \ifpackageloaded{babel}{% if babel is loaded
90   \addto\captionsitalian{\italiandictionary}%
91 }{%else if polyglossia is loaded
92   \ifpackageloaded{polyglossia}{%
93     \addto\captionsitalian{\italiandictionary}%
94   }{%
95 }%
96 \newcommand{\@durationlabeldefault}{\@verifichedurationlabeldefault}%
97 \newcommand{\@asyyearlabeldefault}{\@verifichesyyearlabeldefault}%
98 \newcommand{\headercandidatename}{\@verifichheadercandidatename}%
99 \newcommand{\headerclasslabel}{\@verifichheaderclasslabel}%
100 \newcommand{\headerdatelabel}{\@verifichheaderdatelabel}%
101 \newcommand{\exerciselabel}{\@verifichexerciselabel}%
102 \newcommand{\solutionlabel}{\@verifichsolutionlabel}%
```

7.8 Macro and environment

Global variables The package defined some global variable, that are used to indicate the institute, the year, the duration of the test, its type, the instructions and the solar year.

```
103 \newcommand{\institute}[1]{%
104   \gdef\@institute{#1}}
105 \newcommand{\duration}[2][\@durationlabeldefault]{%
106   \gdef\@durationpreamble{#1}\gdef\@duration{#2}%
107   \showinstructiondelimitertrue}%
108 \newcommand{\testtype}[1]{%
109   \gdef\@testtype{#1}}
```

```

110 \newcommand{\subtitle}[1]{%
111     \gdef\@subtitle{#1}}%
112 \newcommand{\instruction}[1]{%
113     \gdef\@instruction{\ignorespaces#1}%
114     \showinstructiondelimitertrue}
115 \newcommand{\asyear}[2][\@asyearlabeldefault]{%
116     \gdef\@asyearpreamble{#1}\gdef\@asyear{#2}}

```

Fonts and aspects One of the feature of this package is the flexibility about the aspect of the final result. All the main font or typographic elements (like instructions delimiters or the difficulty symbol) are defined as macro.

```

117 \newcommand{\headerfont}[1]{%
118     \sffamily\color{darkgray}#1}
119 \newcommand{\instituteFont}[1]{%
120     \large \sffamily\color{darkgray}#1}
121 \newcommand{\asyearfont}[1]{%
122     \hspace*{\fill} #1\hspace*{\fill}}
123 \newcommand{\testtypefont}[1]{%
124     \bgroup\hspace*{\fill}\scshape\huge#1\hspace*{\fill}\egroup}
125 \newcommand{\subtitledfont}[1]{%
126     \bgroup\hspace*{\fill}\Large\scshape#1\hspace*{\fill}\egroup}%
127 \newcommand{\instructiondelimiter}{%
128     \hrulefill}
129 \newcommand{\instructionfont}[1]{%
130     \sffamily#1}
131 \newcommand{\durationfont}[1]{%
132     \bgroup\hspace*{\fill}\@durationpreamble\space\bfseries #1\hspace*{\fill}\egroup}
133 \newcommand{\@header@candidatename}{%
134     \noindent\headercandidatename\label%
135     \space\underline{\hspace{\candidatename\ruledlength}}}
136 \newcommand{\@header@class}{%
137     \headerclasslabel\space%
138     \underline{\hspace{\class\ruledlength}}}
139 \newcommand{\@header@date}{%
140     \headerdatelabel\space%
141     \underline{\hspace{\date\ruledlength}}}
142 \newcommand{\@header}{%
143     \@header@candidatename\hfill\@header@class\hfill\@header@date}
144 \newcommand{\solutionfont}{%
145     \footnotesize%
146     \ifsolutionscolor\color{\solutionscolor}\fi%
147     \itshape}
148 \newcommand{\diffsymb}{%
149     $\bigstar$}
150 \newcommand{\closedquestionitem}{%
151     $\square$}
152 \newcommand{\@ptprefix}{%
153     \underline{\hspace{\ptruledlength}}/}
154 \newcommand{\ptprefix}[1]{%
155     \renewcommand{\@ptprefix}{#1}}

```

```

156 \newcommand{\@partialptprefix}{}
157 \newcommand{\partialptprefix}[1]{%
158   \renewcommand{\@partialptprefix}{#1}}
159 \newcommand{\@ptsinglabel}{pt}
160 \newcommand{\@ptplurlabel}{pt}
161 \NewDocumentCommand{\ptlabel}{o m}{%
162   \renewcommand{\@ptplurlabel}{#2}%
163   \renewcommand{\@ptsinglabel}{\IfNoValueTF{#1}{#2}{#1}}}%
164 \newcommand{\@useptlabel}[1]{%
165   \ifnum#1=1 \@ptsinglabel \else \@ptplurlabel\fi}
166 \newcommand{\@partialptsinglabel}{pt}
167 \newcommand{\@partialptplurlabel}{pt}
168 \NewDocumentCommand{\partialptlabel}{o m}{%
169   \renewcommand{\@partialptplurlabel}{#2}%
170   \renewcommand{\@partialptsinglabel}{\IfNoValueTF{#1}{#2}{#1}}}%
171 \newcommand{\@usepartialptlabel}[1]{%
172   \ifnum#1=1 \@partialptsinglabel \else \@partialptplurlabel\fi}
173 \newcommand{\@lpartialpt@delimiter}{(}
174 \newcommand{\@rpartialpt@delimiter}{)}
175 \newcommand{\@definepartialptdelimiter}[2]{
176   \renewcommand{\@lpartialpt@delimiter}{#1}
177   \renewcommand{\@rpartialpt@delimiter}{#2}}
178 \newcommand{\partialptdelimiters}[1]{%
179   \@definepartialptdelimiter#1}
180 \newcommand{\@lpt@delimiter}{(}
181 \newcommand{\@rpt@delimiter}{)}
182 \newcommand{\@defineptdelimiter}[2]{%
183   \renewcommand{\@lpt@delimiter}{#1}%
184   \renewcommand{\@rpt@delimiter}{#2}}
185 \newcommand{\ptdelimiters}[1]{%
186   \@defineptdelimiter#1}
187 \newcommand{\checkmarker}{$\checkmark$}%
188 \newcommand{\verifiche@checkmarker}{%
189 \bgroup%
190 \ifsolutionscolor\color{\solutionscolor}\fi%
191 \checkmarker%
192 \egroup}%
193 % \end{macrocode}
194 %\iffalse
195 %% convert ref to num
196 %\fi
197 % \begin{macrocode}
198 \newcommand*\@convertreftonum}[1]{%
199   \romannumeral
200   \@ifundefined{r@#1}{%
201     \expandafter\ltx@zero
202     \rc@default}{%
203     \expandafter\expandafter\expandafter\rc@extract@
204     \expandafter\expandafter\expandafter!%
205     \csname r@#1\expandafter\endcsname
206     \expandafter{\rc@default}\@nil}%
207 }

```

```

208 \def\rc@default{0}%
209 \long\def\rc@extract@#1#2#3\@nil{%
210     \ltx@zero
211     #2}
212 \chardef\ltx@zero=0%

213 \newcommand{\verifich@initremainingvspace}{%
214     \pgfmathsetlength{\verifich@remainingvspace}{%
215         floor((\dimexpr\pagegoal-\pagetotal-.5\baselineskip\relax)/\spacedim)*\spacedim%
216     }%
217 }%

218 \newcommand{\verifich@initremaininghspace}{%
219     \pgfmathsetlength{\verifich@remaininghspace}{%
220         floor(\linewidth/\spacedim)*\spacedim%
221     }%
222 }%

```

Page styles The package define two new pagestyle: `mainverifich` and `verifich`. The former is used if `\@institute` is defined (or if `noinstitute` is not declared) as `\thispagestyle` mode. The latter can be used by the user and print, on the odd pages, the name, date and class of the candidate.

```

223 \newcommand{\ps@mainverifich}{%
224     \renewcommand\@oddhead{%
225         \parbox{\textwidth}{\centering \institute\font{\@institute}}%
226     }%
227     \renewcommand\@evenhead{}%
228     \renewcommand{\@oddfoot}{\hfill\normalfont\textrm\thepage\hfill}%
229     \let\evenfoot\@oddfoot%
230 }

231 \newcommand{\ps@verifich}{%
232     \renewcommand\@oddhead{
233         \ifodd\thepage\parbox{\textwidth}{\headerfont\@header}\fi}%
234     \renewcommand{\@oddfoot}{\hfill\normalfont\textrm\thepage\hfill}%
235     \let\evenfoot\@oddfoot%
236 }

```

Print the heading The purpose of `\printhead` macro is to typeset the headings of the page, printing the name of the institute (if included), the type of the test (if included) and so on.

```

237 \newcommand{\printhead}{%
238     {\ifdefined\@institute\ifinstitute\thispagestyle{mainverifich}\fi\else\thispagestyle{pl
239     {\ifcandidatename\@header\par\vspace{\baselineskip}\fi}%
240     {\ifdefined\@asyear\ifasyear\asyearfont{\@asyearpreamble\space\@asyear}\par\vspace{\basel
241     {\ifdefined\@testtype\iftesttype\testtypefont{\@testtype}\par\fi\fi}%
242     {\ifdefined\@subtitle\ifsubtitle\vspace{\baselineskip}\subtitlefont{\@subtitle}\par\fi\fi
243     {\ifshowinstructiondelimiter\noindent\instructiondelimiter\par\fi}%
244     {\ifdefined\@instruction\ifinstruction\instructionfont{\noindent\@instruction}\par\fi\fi
245     {\ifdefined\@duration\ifduration\durationfont{\@duration\par\fi\fi}%
246     {\ifshowinstructiondelimiter\noindent\instructiondelimiter\par\fi}%
247 }

```

Esercizio. The main element of the package is the `eserciz` environment. It accepts some options define via `xkeyval` package. The element that are reproduced

in the left margin are stored into a box and produce if `showmargin` boolean variable is switch to *true*.

```

248 \define@key{esercizio}{label}[\exerciselabel]{%
249   \def\tempeserciziolabel{#1}}%
250 \define@key{esercizio}{diff}[none]{
251   \def\diffstar{\hfill\foreach \i in {1,..., #1}{\diffsymsb}}%
252   \showmarginexercisetrue}%
253 \define@key{esercizio}{pt}[none]{
254   \ifshowexercisep%
255   \def\pointes{#1}%
256   \def\espoint{%
257     \hfill\@ptprefix\@lpt@delimiter%
258     \pointes\space\@useptlabel{\pointes}%
259     \@rpt@delimiter}
260   \showmarginexercisetrue
261   \fi}%
262 \define@key{esercizio}{partialpt}[none]{%
263   \ifshowexercisep%
264   \def\@totalpartialpt{\convertreftonum{ptes@\theexercisenum}}%
265   \def\sumpartialpt{%
266     \@ptprefix\@lpt@delimiter%
267     \@totalpartialpt\space\@useptlabel{\@totalpartialpt}%
268     \@rpt@delimiter}%
269   \showmarginexercisetrue
270   \fi}%

```

The part of the exercise (i.e. the title and the margin note) are stored into macro in order to streamline the code.

```

271 \newcommand{\exercisetitle}{%
272   \ifdefined\tempeserciziolabel\tempeserciziolabel%
273   \else \exerciselabel\fi%
274 }
275 \newsavebox{\box@marginexercise}
276 \newcommand{\exercisemargin}{%
277   \sbox{\box@marginexercise}{\parbox[t]{\marginparwidth}{%
278     \ifdefined\diffstar\hfill\diffstar\par\fi% stelle di difficoltà
279     \ifdefined\espoint\hfill\espoint\par\fi%[-\baselineskip] %punteggio
280     \ifdefined\sumpartialpt\hfill\sumpartialpt\fi} %punteggio parziale automatico
281   }%
282   \reversemarginpar\marginpar{\usebox{\box@marginexercise}}%
283 }%

```

If the exercise admit points, they are stored in the `.aux` file of the main document in order to be recalled via `\ref` command.

```

284 \newcommand{\savepointsaux}{%
285   \ifdefined\sumpartialpt%
286     \protected@write \@auxout {}{\string\newlabel {ptes@\theexercisenum}}{\thepartialp
287     \fi%
288     \ifdefined\espoint%
289       \protected@write \@auxout {}{\string\newlabel {ptes@\theexercisenum}}{\pointes}{}}
290     \fi%
291 }%

```

`esercizio` (*env.*) The content of the exercise are managed by the `esercizio` environment. For every `esercizio` the counter `exercisenumber` is increased, the `showmarginexercise` are reset to false (to avoid the generation of spurious margin note if unnecessary) and the keys are set. After that, the code produces the label of the environment, the body and manages the space before e after the environment in the page. If the exercise presents some scores, they are stored via `savepointsaux`.

```

292 \NewDocumentEnvironment{esercizio}{o +b}{%
293   \refstepcounter{exercisenumber}%
294   \showmarginexercisefalse%
295   \IfValueT{#1}{\setkeys{esercizio}{#1}}%
296   \par\addvspace{\baselineskip}%
297   {\noindent\bfseries\exercisetitle\ifexercisenumbered\space\theexercisenumber\fi}%
298   \ifshowmarginexercise\exercisemargin\fi%
299   \par% newline after label
300   \setlength\parindent{0pt}%no parindent
301   \nobreak\ignorespaces#2%body of the environment
302 }{%
303   \par%
304   \ignorespacesafterend%
305   \savepointsaux%
306 }%
```

`esercizio*` (*env.*) The `esercizio*` environment has the same structure of the previous one, except for the code used to show the number of the exercise.

```

307 \NewDocumentEnvironment{esercizio*}{o +b}{%
308   \showmarginexercisefalse%
309   \IfValueT{#1}{\setkeys{esercizio}{#1}}%
310   \par\addvspace{\baselineskip}{\noindent\bfseries\exercisetitle}%
311   \ifshowmarginexercise\exercisemargin\fi%
312   \par% newline after label
313   \setlength\parindent{0pt}%no parindent
314   \nobreak\ignorespaces#2% body of the environment
315 }{%
316   \par%
317   \ignorespacesafterend
318   \savepointsaux%print partial pt sum
319 }%
```

`\partialpt` The `partialpt` macro define the content of the margin note where it's used.

```

320 \NewDocumentCommand{\partialpt}{m}{%
321   \ifshowexercisep%
322   \leavevmode\reversemarginpar\marginpar{%
323     \hfill\@partialptprefix%
324     \@lpartialpt@delimiter#1\space\@usepartialptlabel{#1}%
325     \@rpartialpt@delimiter%
326     \addtocounter{partialpoints}{#1}\ignorespaces%
327     \fi}%
```

Soluzione

`soluzione` (*env.*) The main element for the "solutions" mode is the `soluzione` environment.

```

328 \define@key{soluzione}{label}[\solutionlabel]{%
329   \def\verifiche@tempcsoluzionelabel{#1}%
```



```

330 }
331 \newcommand{\verifiche@solutionlabel}{%
332   \ifdefined\verifiche@temp soluzione label\verifiche@temp soluzione label%
333   \else\solutionlabel\fi%
334 }%
335 \NewDocumentEnvironment{soluzione}{o +b}{%
336   \IfValueT{#1}{\setkeys{soluzione}{#1}}%
337   \ifsol%
338     \par\advspace{\baselineskip}%
339     {\noindent\bfseries%
340     \ifsolutionscolor\color{\solutionscolor}\fi%
341     \verifiche@solutionlabel}%
342     \par\nobreak\noindent\ignorespaces%
343     \setlength\parindent{0pt}%no parindent
344     \solutionfont #2%
345   \fi%
346 }{%
347   \ifsol%
348     \par\ignorespacesafterend
349   \fi%
350 }%

```

`\inlinesol` The code inside `\inlinesol` is composed only if in "solutions" mode.

```

351 \newcommand{\inlinesol}[1]{%
352   \ifsol%
353     \bgroup%
354       \ifmode\mathit{\solutionfont#1}%
355       \else{\solutionfont#1}%
356     \fi%
357   \egroup%
358   \else\ignorespaces%
359   \fi%
360 }

```

`\completetext` The `\completetext` print a horizontal line on which write the solution if the "solutions" mode is active.

```

361 \newcommand{\completetext}[2][4.5cm]{%
362   \ifsol%
363     $\overset{\displaystyle\text{\solutionfont#2}}{\underline{\hspace{#1}}}$%
364   \else%
365     $\overset{\relax}{\underline{\hspace{#1}}}$%
366   \fi%

```

`\truefalse` The `\truefalse` is generated via `\tikz` macro.

```

367 \newcommand{\truefalse}[1]{%
368   %True
369   \framebox{\parbox[c][\verifiche@truefalsesquares]{\verifiche@truefalsesquares}{\centering
370   \ifsol\llap{%
371     \if#1@\verifiche@Tletter%
372     \Large\verifiche@checkmarker\hspace{-1pt}%
373   \fi}%
374   \fi%
375   \quad%
376   %False

```

```

377 \fbox{\parbox[c][\verifiche@truefalsesquares]{\verifiche@truefalsesquares}{\centering\@ve
378 \ifsol\llap{
379 \if#1\@verifiche@Fletter%
380 \Large\verifiche@checkmarker\hspace{-1pt}%
381 \fi}%
382 \fi%%
383 }

```

`\finderror` The `\finderror` write the wrong text in "test mode" and than superimpose some mark (thanks to `ulem` package) and the correct text in "solutions mode".

```

384 \newcommand{\finderror}[2]{%
385 \ifsol%
386 $\overset{\text{\solutionfont#2}}{\text{\xout{#1}}}$%
387 \else%
388 #1%
389 \fi%
390 }

```

`closedquestion` (*env.*) The `closedquestion` environment define a new list whose item elements are preceded by square symbol, and every item is `\hfill` spaced. The `\checked` symbol is controlled and generated (in the "solutions mode") via `\tikz`.

```

391 \newlist{crocette}{itemize}{1}%
392 \newlist{crocette*}{itemize*}{1}%
393 \setlist[crocette]{label=\closedquestionitem, leftmargin=*, parsep=Opt }%
394 \setlist[crocette*]{label=\closedquestionitem, itemjoin={\hfill}}%
395 \newcommand{\checked}{%
396 \ifsol%
397 \closedquestionitem\llap{\verifiche@checkmarker\hspace{-1pt}}%
398 \else%
399 \closedquestionitem%
400 \fi}
401 \NewDocumentEnvironment{closedquestion}{0}{+b}{%
402 \begin{crocette}[#1]#2%
403 }{%
404 \end{crocette}%
405 }%
406 \NewDocumentEnvironment{closedquestion*}{0}{ +b}{%
407 \begin{crocette*}[#1]#2%
408 }{%
409 \end{crocette*}%
410 }%

```

`\openquestion` The `\openquestion` use key-value options defined as following presented.

```

411 \define@key{openquestion}{spacedim}[0.5cm]{\setlength{\spacedim}{#1}\verifiche@initremainingv
412 \define@key{openquestion}{height}[\the\verifiche@remainingvspace]{\def\openquestionheight{#1}}
413 \define@key{openquestion}{width}[\the\verifiche@remaininghspace]{\def\openquestionwidth{#1}}%
414 \define@key{openquestion}{linecolor}[gray]{\def\openquestionlinecolor{#1}}%
415 \define@choicekey**{openquestion}{type}[\val\nr]{lines, squares}[squares]{%
416 \ifcase\nr\relax%
417 % lines
418 \openquestionsquaredfalse\openquestionlinestrue%
419 \or%
420 % squares
421 \openquestionsquaredtrue\openquestionlinesfalse%

```

```

422 \fi}%
423 {\PackageError{verifiche}{Errore nell'opzione openquestion}{Inserire lines o squares}}

```

The macro starts with an initiation of predefined type of `\openquestion`, than possible options are overloaded. Depending on the type of the `\openquestion`, squared or ruled background is composed and in the foreground is reproduce the text of the argument of the command (in the "solutions mode").

```

424 \NewDocumentCommand{\openquestion}{o m}{%
425 \setkeys{openquestion}{type, spacedim, width, height, linecolor} %init
426 \IfValueT{#1}{\setkeys{openquestion}{#1}}%
427 \vspace{-\baselineskip}%
428 \begin{center}
429 \ifopenquestionsquared% if squared open question
430 \begin{tikzpicture}%
431 \draw[step=\the\spacedim, \openquestionlinecolor]%
432 (0,0) grid (\openquestionwidth,-\openquestionheight);%
433 \node[%
434 anchor=base west,%
435 align=left, text width=\openquestionwidth,%
436 inner sep=0cm,%
437 font=\solutionfont,%
438 execute at begin node=\setlength{\baselineskip}{\the\spacedim}]%
439 at (0, -\the\spacedim){\ifsol#2\fi};
440 \end{tikzpicture}%
441 \else% if lined open question
442 \begin{tikzpicture}%
443 \pgfmathparse{int(\openquestionheight/\the\spacedim)}%
444 \foreach \y in {1, ..., \pgfmathresult}%
445 \draw[\openquestionlinecolor] (0, -\y*\the\spacedim) -- +(\openquestionwidth,0);
446 \node[%
447 anchor=base west,align=left,%
448 text width=\openquestionwidth,%
449 inner sep=0cm, font=\solutionfont,%
450 execute at begin node=\setlength{\baselineskip}{\the\spacedim}]%
451 at (0, -\the\spacedim){\ifsol#2\fi};%
452 \end{tikzpicture}%
453 \fi}%
454 \end{center}%
455 }

```

`\textandimage` The `\textandimage` manages two column material via `minipage` environment.

```

456 \NewDocumentCommand{\textandimage}{0{.7\textwidth} 0{.3\textwidth} m m}{%
457 \begin{minipage}{#1}
458 {#3}
459 \end{minipage}
460 \hfill
461 \begin{minipage}{#2}
462 {#4}
463 \end{minipage}
464 }

```

`\makecolumn` The `\makecolumn` give the possibility to create multiple column in the document.

```

465 \NewDocumentCommand{\makecolumn}{0{t} m +m}{%
466 \parbox[#1]{#2}{#3}%

```

```

467     \@ifnextchar\makecolumn{\hspace*{\columnseparation}}{ }%
468 }%

```

`multitest` (*env.*) The environment `multitest` handle the possibility of generating variation of the same test.

```

469 \NewDocumentEnvironment{multitest}{0{1} +b}{
470     \newcounter{multitestcounter}
471     \toks1={\foreach \i in {1,...,#1}}
472     \the\toks1{%
473         \setcounter{multitestcounter}{\i}%
474         \setcounter{page}{1}%
475         \setcounter{exercisenum}{0}%
476         \pgfmathsetseed{\i}%
477         #2%
478         \clearpage%
479     }%
480 }{}

```

Change History

v1.0		noexercisep global option to hide the score of each exercise. . . 1
	General: Initial version. 1	
v2.0		v4.4
	General: Better performance in exercise environment. 1	General: Minor update. Fixed labels issues. 1
v2.2		v5.0
	General: Modify reference to points of an exercise. 1	General: Major update. Remove some unnecessary packages. Remove spurious space in closedquestion environment. Add a starred version of closedquestion environment. Add the possibility to add local label to soluzione environment. Add a new command: makecolumn. 1
v3.0		v5.1
	General: Remove fancyhdr and introduce a new page style. Improve the printheadng command. 1	General: Minor update. Clean the code for printheadng command; better appearance for openquestion command. . . . 1
v4.0		v5.2
	General: Introduce comando to change the aspect of the exercise score. 1	General: Minor update. Resolve bugs in printheadng command. 1
v4.1		
	General: Introduce ptrulerlength to change the aspect of the exercise score. 1	
v4.2		
	General: Solved truefalse issue in solutionsmode. Add	

<code>\exercisenumberedtrue</code>	<code>\institutefalse</code> ... 37	<code>\ps@mainverifiche</code> . 223
..... 31	<code>\institutefont</code>	<code>\ps@verifiche</code> 231
<code>\exercisetitle</code> 12, 119, 225	<code>\ptdelimiters</code> .. 13, 185
.... 271, 297, 310	<code>\institutetrue</code> 21	<code>\ptlabel</code> 14, 161
<code>\expandafter</code> .. 201,	<code>\instruction</code> 4, 112	<code>\ptprefix</code> 13, 154
203, 204, 205, 206	<code>\instructiondelimiter</code>	<code>\ptrulerlength</code>
F	. 12, 127, 243, 246 56, 65, 153
<code>\finderror</code> 10, 384	<code>\instructionfalse</code> . 38	Q
<code>\framebox</code> 369	<code>\instructiontrue</code> .. 22	<code>\quad</code> 375
H	<code>\instrunctionfont</code> .	S
<code>\headercandidatename</code> 12, 129, 244	<code>\savepointsaux</code>
<code>\headerclasslabel</code> .	<code>\italiandictionary</code> 284, 305, 318
..... 13, 98, 134 78, 90, 93	<code>\setkeys</code> 295,
<code>\headerdatelabel</code> ..	<code>\item</code> 9	309, 336, 425, 426
.. 13, 13, 100, 140	L	<code>\showexercisepfalse</code> 45
<code>\headerfont</code> 12, 117, 233	<code>\long</code> 209	<code>\showexerciseptrue</code> 33
I	M	<code>\showinstructiondelimiterfalse</code>
<code>\if</code> 371, 379	<code>\makecolumn</code> 11, 465 29
<code>\ifasyear</code> 24, 240	<code>\multitest (env.)</code> 15, 469	<code>\showinstructiondelimitertrue</code>
<code>\ifcandidatename</code> 26, 239	<code>\multitestcounter</code> . 16 107, 114
<code>\ifcase</code> 416	O	<code>\showmarginexercisefalse</code>
<code>\ifduration</code> 23, 245	<code>\openquestion</code> .. 10, 411 32, 294, 308
<code>\ifexercisesnumbered</code>	<code>\openquestionheight</code>	<code>\showmarginexercisetrue</code>
..... 31, 297 412, 432, 443 252, 260, 269
<code>\iffalse</code> 194	<code>\openquestionlinecolor</code>	<code>\soltrue</code> 36
<code>\ifinstitute</code> ... 21, 238 414, 431, 445	<code>\solutionfont</code> 12, 144,
<code>\ifinstruction</code> . 22, 244	<code>\openquestionlinesfalse</code>	344, 354, 355,
<code>\ifNoValueTF</code> .. 163, 170 34, 421	363, 386, 437, 449
<code>\ifnum</code> 165, 172	<code>\openquestionlinestrue</code>	<code>\solutionlabel</code>
<code>\ifodd</code> 233 418	. 14, 102, 328, 333
<code>\ifopenquestionlines</code> 34	<code>\openquestionsquaredfalse</code>	<code>\solutionscolor</code> ...
<code>\ifopenquestionsquared</code> 35, 418	. 46, 146, 190, 340
..... 35, 429	<code>\openquestionsquaredtrue</code>	<code>\solutionscolorfalse</code> 25
<code>\ifprintedheading</code> . 30 421	<code>\solutionscolortrue</code> 46
<code>\ifshowexercisep</code> .	<code>\openquestionwidth</code> .	<code>soluzione (env.)</code> . 8, 328
. 33, 254, 263, 321 413,	<code>\square</code> 151
<code>\ifshowinstructiondelimiter</code>	432, 435, 445, 448	<code>\subtitle</code> 5, 110
..... 29, 243, 246	P	<code>\subtitlefalse</code> 43
<code>\ifshowmarginexercise</code>	<code>\pagegoal</code> 215	<code>\subtitlefont</code> . 125, 242
..... 32, 298, 311	<code>\pagetotal</code> 215	<code>\subtitltrue</code> 28
<code>\ifsol</code> 20,	<code>\partialpt</code> 7, 320	<code>\sumpartialpt</code>
337, 347, 352,	<code>\partialptdelimiters</code> 265, 280, 285
362, 370, 378, 13, 178	T
385, 396, 439, 451	<code>\partialptlabel</code> 14, 168	<code>\tempeserciziolabel</code>
<code>\ifsolutionscolor</code> .	<code>\partialptprefix</code> 13, 157 249, 272
. 25, 146, 190, 340	<code>\pgfmathsetlength</code> .	<code>\testtype</code> 4, 108
<code>\ifsubtitle</code> 28, 242 214, 219	<code>\testtypefalse</code> 42
<code>\iftesttype</code> 27, 241	<code>\pgfmathsetseed</code> ... 476	<code>\testtypefont</code>
<code>\inlinesol</code> 8, 351	<code>\pointes</code> .. 255, 258, 289 12, 123, 241
<code>\institute</code> 4, 103	<code>\printedheadingfalse</code> 30	<code>\testtypetrue</code> 27
	<code>\printheadng</code> ... 5, 237	<code>\text</code> 363, 386

