The schl package[*]

Tassos Tsesmetzis

`ttsesmetzis@gmail.com`

November, 2019

Abstract

schl is a X∃LATEX package that provides commands and environments suitable for document types that appear in a classroom enviromnent. It's development is based on the Greek school practice, but it may be usefull in other contexts also.

1 Introduction

Worksheets and tests are common document types in a classroom. schl package comes with macros that facilitate the creation of these documents. It has list environments for questions, exercises and tasks. Other environments of the package can be used for tickable or multiple choice answers. There are also commands for typesetting solutions, hints and answers to exercises.

Furthermore, you can set the name of the teacher, subject, grade, headmaster, school, date, school year and use these to print school's logo or information about an exam. schl has commands to typeset headers for each document type, a macro for typing the points of an exercise and two commands for blank space. There is also a macro for typesetting a wish for good luck!

By default, schl prints all macros that accept text as `undefined`. As of this version (v1.0), Greek is the only supported language. You can set it with the option `greek`. Other languages can be supported by redefining package's internal macros.

schl loads the packages `fontspec`, `enumitem`, `mdframed` and `amsmath`. It is written for X∃LATEX, but can be used by any system that supports `fontspec`.

2 Macros

2.1 Blank space

`\lowerdots`
`\blankspace`
Usually, we need to designate blank space in a document. schl package has two

---

[*]This document corresponds to schl v1.0, dated 2020/06/26.

commands for this. The first one \lowerdots [⟨*length*⟩]{⟨*number*⟩}, prints ⟨*number*⟩ dots. Optional argument ⟨*length*⟩ sets the deviation from base line. It's default value is −0.3ex.

```
1 \newcommand\lowerdots[2][-0.3ex]{%
2   \begingroup
3   \lccode`m=`.\relax
4   \raisebox{#1}{\lowercase\expandafter{\romannumeral\number\number#2 000}}%
5   \endgroup
6 }
```

\blankspace [⟨*length*⟩]{⟨*linelength*⟩} prints a line with length ⟨*linelength*⟩. The optional argument is the deviation from the base line and it's default value is −0.3ex. \schl@rulethickness is the default thickness for all \blankspace lines.

```
7 \newcommand\blankspace[2][-0.3ex]{%
8   \raisebox{#1}{\rule{#2}{\schl@rulethickness}}
9 }
```

2.2   Lists

schl package defines seven types of lists. These are question, exercise, schltask, multichoice, tickchoice, truefalse and matchique. tickchoice comes also with a stared version tickchoice*. All of them depend the package enumitem.

question   These environments are enumerate-like lists. List's \item is of the form ⟨*type*⟩
exercise   ⟨*counter*⟩, where type is \question@term for question, \exercise@term for exercise
schltask   and \task@term for schltask. ⟨*counter*⟩ is the internal counter of the environment.

```
10 \newlist{question}{enumerate}{1}
11 \setlist*[question]{%
12   align=left,
13   label=\bf \question@term\  \arabic*.,
14   wide,
15   leftmargin=0pt,
16   labelindent=0pt
17 }

18 \newlist{exercise}{enumerate}{1}
19 \setlist*[exercise]{%
20   align=left,
21   label=\bf\exercise@term\  \arabic*.,
22   wide,
23   leftmargin=0pt,
24   labelindent=0pt
25 }

26 \newlist{schltask}{enumerate}{1}
27 \setlist*[schltask]{%
28   align=left,
29   label=\bf\letterspace{\defaultletterspace}\task@term\  \arabic*,
30   wide,
31   leftmargin=0pt,
32   labelindent=0pt
```

33 }

\letterspace   The macro \letterspace{⟨*number*⟩} is used to set the horizontal space of adjacent characters in a word. It is based on the \addfontfeature macro from the package fontspec. The argument ⟨*number*⟩ is a percentage of the font size. In schl package is used to set the space between capital word letters.

34 \def\letterspace#1{\addfontfeature{LetterSpace=#1}}

multichoice   The multichoice environment is used to typeset multiple choice answers.

35 \newlist{multichoice}{enumerate*}{1}
36 \setlist*[multichoice]{
37   labelindent=\parindent,
38   label=\Alph*.,
39   itemjoin=\hspace{\fill},
40   before=\hspace{\fill},
41   after=\hspace{\fill}
42 }

tickchoice   The environments tickchoice and tickchoice* are variants of the itemize list.
tickchoice*   For both cases, each item is preceded by a square. tickchoice stacks items vertically,

43 \newlist{tickchoice}{itemize}{1}
44 \setlist[tickchoice]{labelindent=\parindent,label={\large$\square$}}

while tickchoice* stacks them horizontally.

45 \newlist{tickchoice*}{itemize*}{1}
46 \setlist*[tickchoice*]{
47   labelindent=\parindent,
48   label={\large$\square$},
49   itemjoin=\hspace{\fill},
50   before=\hspace{\fill},
51   after=\hspace{\fill}
52 }

truefalse   truefalse is a variant of the enumerate environment. Each \item is divided in two parts. The first part is the text that follows the \item macro. The second part is a \parbox that prints \trueabbr@term and \falseabbr@term.

53 \newlist{truefalse}{enumerate}{1}
54 \setlist[truefalse]{label={\bf \arabic*.},%
55   before*={%
56   \let\defaultitem\item%   Save the standard definition of \item in a macro.
57   \toggletrue{first}%   Set the first toggle with initial value true.
58   \def\item{%
59   \iftoggle{first}{%
60   \togglefalse{first}%   Set the first toggle to take the value false.
61   \defaultitem \begin{minipage}[t]{0.8\linewidth minus \truefalselength}%
62   }{%
63   \end{minipage}\hfill\truefalselabel\defaultitem%
64   \begin{minipage}[t]{0.8\linewidth minus \truefalselength}%
65   }
66   }% new, temporary defition of \item
67   },

```
68    after*={%    This takes care of adding the fill for the final item on
69       %          the list and just makes sure that \item is reset to its standard definition
70     \end{minipage}\hfill\truefalselabel% fill for final item in list
71     \let\item\defaultitem% restore standard definition of \item
72   }%
73 }
```

truefalse*   `truefalse*` is based on the `truefalse` environment. A line with `\trueabbr@term`
and `\falseabbr@term` at the end prepends the list of items. In contrast with the
`truefalse` environment, each `\item` ends with a pair of squares.

```
74 \newlist{truefalse*}{enumerate}{1}
75 \setlist[truefalse*]{label={\bf \arabic*.},%
76   before*={%
77     \strut\hspace{\leftmargin}\hfill\truefalselabel\hspace{\rightmargin}\strut\vspace{-1ex}% Pr
78     \let\defaultitem\item%      Save the standard definition of \item in a macro.
79     \toggletrue{first}%                Set the first toggle with initial value true.
80     \def\item{%
81       \iftoggle{first}{%
82         \togglefalse{first}%            Set the first toggle to take the value false.
83         \defaultitem\begin{minipage}[t]{0.8\linewidth minus \truefalselength}%
84         }{%
85         \end{minipage}\hfill\truefalsesquares\defaultitem%
86         \begin{minipage}[t]{0.8\linewidth minus \truefalselength}%
87         }
88       }% new, temporary defition of \item
89     },
90     after*={%    This takes care of adding the fill for the final item on
91       %          the list and just makes sure that \item is reset to its standard definition
92     \end{minipage}\hfill% fill for final item in list
93     \truefalsesquares
94     \let\item\defaultitem% restore standard definition of \item
95   }%
96 }
```

matchingque   The macro `\matchingque{⟨CSV⟩}{⟨CSV⟩}` is used to typeset matching questions.
⟨CSV⟩ are comma separated values. The ⟨CSV⟩s of the first argument are the parts of
the matching questions that will be print in the left column. Similarly, the ⟨CSV⟩ of the
second argument are going to be printed on the right column of the matching questions.

```
97 \newcommand\matchingque[3][300pt]{%
98   \begin{center}
99     \parbox[c]{#1}{
100       \parbox[c]{\leftmatchwidth}{%
101         \begin{leftmatching}
102           \@for\tmp:=#2%
103           \do{%
104           \item \tmp
105           }
106         \end{leftmatching}
107       }
108       \hfill%
```

```
109      \parbox[c]{\rightmatchwidth}{%
110        \begin{rightmatching}
111          \@for\tmp:=#3%
112          \do{%
113          \item \tmp
114          }
115        \end{rightmatching}
116      }
117    }
118  \end{center}
119 }
```

leftmatching Environments `leftmatching` and `rightmatching` are used to typeset each column
rightmatching in `\matchingque`.

```
120 \newlist{leftmatching}{enumerate}{1}
121 \newlist{rightmatching}{enumerate}{1}
122 \setlist*[leftmatching]{label=\bf\Alph*.}
123 \setlist*[rightmatching]{label=\bf\arabic*.}
```

2.3 Answers, solutions and hints

\answer Macro `\answer{`⟨*text*⟩`}` prints (`\answerabbr@term \meta{text}`) at the right
end of the current line.

```
124 \newcommand\answer[2][\hfill\footnotesize]{%
125   {#1 (\answerabbr@term: #2)}
126 }
```

\solution Macro `\solution{`⟨*text*⟩`}` is used to typeset the solution of an exercise.

```
127 \newcommand\solution[1]{%
128   \par\noindent\strut\hfill\textbf{\solution@term}\hfill\strut\par%
129   \noindent #1
130 }
```

\hint schl provides the macro `\hint{`⟨*text*⟩`}` for typesetting exercise hints.

```
131 \newcommand\hint[2][\par\noindent\footnotesize]{%
132   {#1\textbf{\hint@term:} #2}%
133 }
```

\deadline A feature of homework assignments is a deadline date. `\deadline{`⟨*date*⟩`}` prints
`\deadline@term` followed by argument ⟨*date*⟩.

```
134 \newcommand\deadline[2][\noindent\bf]{%
135   {#1\deadline@term}: #2%
136 }
```

\remark Add a remark in a document. `\remark{`⟨*text*⟩`}` prints `\remark@term` followed by
argument ⟨*text*⟩.

```
137 \newcommand\remark[2][\noindent\bf]{%
138   {#1\remark@term}: #2%
139 }
```

\reminder   Add a reminder in a document. `\reminder{⟨text⟩}` prints `\reminder@term` followed by argument ⟨*text*⟩.

```
140 \newcommand\reminder[2][\noindent\bf]{%
141   {#1\reminder@term}: #2%
142 }
```

2.4   Titles and headers

\heading   Common document types in a school environment are the worksheet, various tests and final written exams. The macro `\heading{⟨text⟩}` gives a generic header for all these documents.

```
143 \newcommand\heading[1]{%
144   \begin{center}
145     {\bf\large #1}
146   \end{center}
147 }
```

\worksheethd   Macro `\worksheethd{⟨text⟩}` sets the title of a worksheet. It appends ⟨*text*⟩ to `\worksheet@term`.

```
148 \newcommand\worksheethd[1]{%
149   \heading{\worksheet@term\  #1}
150 }
```

\examhd   `\examhd[⟨text⟩]{⟨text⟩}` is used to set the title of tests. The optional argument has the default value `\termtest@term`.

```
151 \newcommand\examhd[2][\termtest@term]{%
152   \heading{#1 #2}
153 }
```

\finalexamhd   Titles for end year exams have a standardized form in Greek schools. `\exams@term` is followed by information about the exam. Then comes `\period@term` with the exam period after it. `\finalexamhd{⟨info⟩}{⟨period⟩}` is used for these cases.

```
154 \newcommand\finalexamhd[2]{%
155   \heading{\letterspace{\defaultletterspace} #1 \exams@term\\[0.5ex] \period@term\  #2}
156 }
```

\schl@framedbox   `\schl@framedbox{⟨text⟩}` prints ⟨*text*⟩ in a centered frame box. It is used by `\theorypart` and `\exercisepart`.

```
157 \newcommand\schl@framedbox[1]{%
158   \begin{center}
159     \fbox{\large{\bf\letterspace{\defaultletterspace} #1} }%
160   \end{center}
161 }
```

\theorypart   Sometimes theory and exercise sections constitute a written test. Macros `\theorypart`
\exercisepart   and `\exercisepart` print headers for those parts.

```
162 \newcommand\theorypart{%
163   \schl@framedbox{\theoryheader@term\!}
164 }
```

and

```
165 \newcommand\exercisepart{%
166     \schl@framedbox{\exerciseheader@term\!}
167 }
```

2.5   School information

\school
\headmaster
\teacher
\subject
\grade
\schoolyear
\setdate
\examtime

The macros \school{⟨*text*⟩}, \headmaster{⟨*name*⟩}, \teacher{⟨*name*⟩}, \subject{⟨*text*⟩}, \grade{⟨*text*⟩}, \schoolyear{⟨*year*⟩}, \setdate{⟨*date*⟩}, \examtime{⟨*time*⟩} define and set the value of internal macros.

```
168 \newcommand\school[1]{\def\schl@school{#1}}
169 \newcommand\headmaster[1]{\def\schl@headmaster{#1}}
170 \newcommand\teacher[1]{\def\schl@teacher{#1}}
171 \newcommand\subject[1]{\def\schl@subject{#1}}
172 \newcommand\grade[1]{\def\schl@grade{#1}}
173 \newcommand\schoolyear[1]{\def\schl@schoolyear{#1}}
174 \newcommand\setdate[1]{\def\schl@date{#1}}
175 \newcommand\examtime[1]{\def\schl@examtime{#1}}
```

\authorities   In a similar vein, \authorities{⟨*text*⟩} is used to define the internal macro \schl@authorities.

```
176 \newcommand\authorities[1]{\def\schl@authorities{#1}}
```

2.6   Other macros for tests

\points   \points[⟨*macro*⟩]{⟨*number*⟩} is used to designate the points of an exercise. {⟨*number*⟩} is the number of points for the current exercise, while [⟨*macro*⟩] can be used to control the space just before the points.

```
177 \newcommand{\points}[2][\hfill\bf\footnotesize]{%
178 {#1(\points@term{#2} #2)}
179 }
```

\fullname   \fullname{⟨*text*⟩} prints \fullname@term followed by ⟨*text*⟩.

```
180 \newcommand\fullname[2][\noindent\rmfamily]{%
181     {#1\fullname@term}: #2%
182 }
```

\datefield   Similarly, \datefield{⟨*text*⟩} prints \date@term with ⟨*text*⟩ after it.

```
183 \newcommand\datefield[2][\noindent\rmfamily]{%
184     {#1\date@term}: #2%
185 }
```

\getdate   \getdate prints \schl@date. The last macro can be set with \setdate.

```
186 \newcommand\getdate{\schl@date}
```

\duration   \duration{⟨*duration*⟩} prints \duration@term with ⟨*duration*⟩ after it.

```
187 \newcommand\duration[2][\noindent\bf]{%
188     {#1\duration@term}: #2%
189 }
```

`\schoollogo`   `\schoollogo{`⟨*width*⟩`}` prints `\schl@school`, `\schl@grade`, `\schl@subject`
and `\schl@teacher`. ⟨*width*⟩ is the length of the `\parbox`.

```
190 \def\schoollogo#1{%
191   \parbox[t]{#1}{%
192     \schl@school\\%
193     \schl@grade\\%
194     \schl@subject\\%
195     \schl@teacher
196   }
197 }
```

`\authoritylogo`   `\authoritylogo[`⟨*number*⟩`]` prints `\sch@authorities` and `\schl@school`. Argument
⟨*number*⟩ is a multiplier for `\baselineskip`. This spaces is added above the macro.

```
198 \newcommand\authoritylogo[1][1.5]{%
199   \noindent\parbox[t][\height]{0.4\textwidth}{%
200     \centering%
201
202     \vspace{#1\baselineskip}
203
204     {\schl@authorities}
205
206     \vspace{3\lineskip}
207
208     {\small\letterspace{\defaultletterspace}\MakeUppercase{\schl@school}}}
209   }
210 }
```

`\signatures`   Some types of written tests end with the names of the headmaster and the teacher(s)
followed by handwritten signatures. `\signatures[`⟨*role*⟩`]{`⟨*signer(s)*⟩`}` prints ⟨*role*⟩.
After it follow the name(s) of the ⟨*signer(s)*⟩, one per line. ⟨*Signer(s)*⟩ is a comma
separated list of values. The default value for ⟨*role*⟩ is `\headmaster@term`. `\signatureslength`
is the length of the `\signatures` block and `\signatureslineskip` is the length
between adjacent lines.

```
211 \newcommand\signatures[2][\headmaster@term]{%
212   \parbox[t]{\signatureslength}{%
213     \setlength \baselineskip{\signaturelineskip}
214     \begin{center}
215       #1%
216       \@for\arg:=#2
217       \do {%
218         \\%
219         \arg%
220       }
221     \end{center}
222   }
223 }
```

`\wish`   `\wish` prints `\schl@wish`, a default wish for tests.

```
224 \newcommand\wish[1][\schl@wish]{%
225   \begin{center}
```

```
226        {\LARGE\bf #1}
227    \end{center}
228 }
```