

ueaexam v4.14: L^AT_EX 2_ε Class for Writing UEA Exam Sheets

Nicola L.C. Talbot

School of Computing Sciences
University of East Anglia
Norwich, Norfolk
NR4 7TJ, United Kingdom.

<http://theoval.cmp.uea.ac.uk/~nlct/>

2012-01-22

Contents

1 Introduction	1
2 Required Packages	2
3 Class Options	3
4 Large Font Sizes for Partially Sighted Students	4
5 Available Commands	5
5.1 Preamble Commands	5
5.2 Document Commands	6
6 Defining Problems and their Solutions in External Files	9
7 Example Documents	11
Index	15

1 Introduction

This is the documentation for ueaexam, a class file for typesetting exam sheets. It is loosely based on the sys-exam package and on the standard article class file¹.

¹although not all the commands defined in the article.cls article are available in this class file

2 Required Packages

The following packages are automatically loaded:

- amsmath
- etoolbox
- probsoln — allows questions and answers to be stored in and loaded from databases. **Must be at least version 3.02. Earlier versions will cause an error.**
- drawwatermark — provides facility to put “DRAFT” or “SOLUTIONS” across the background of each page.
- geometry — sets up page layout.
- ifthen — provides high-level conditional commands.
- calc — used for counter calculations.
- fp — used for adding up the marks for each question.
- graphicx — images can be included using `\includegraphics[<key=value options>]{<filename>}`.
- xcolor — text colour may be changed using `\textcolor{<colour>}{<text>}`.
- enumerate — provides optional argument to enumerate environment to change the item labels.
- paralist — provides inparaenum environment for numbered lists within a paragraph. Note that probsoln also provides textenum for in-line enumerations.
- fmtcount — provides command to convert numerics into words.

If the times class option is specified, the following packages are also loaded:

- mathptmx
- helvet
- courier
- lmodern

The exsizes bundle is required if any of the class options 14pt, 17pt or 20pt are used.

3 Class Options

times Use Times/Helvetica/Courier fonts instead of Computer Modern. (Default.)

notimes Use Computer Modern fonts.

solutions Display solutions given by

`\Solution`

```
\Solution{<text>}
```

and put “SOLUTIONS” banner across every page.

nosolutions Hide solutions given by `\Solution`.

watermarkfirst Only put watermark on first page instead of on all pages. Has no effect if neither solutions nor draft used.

scorecheck Check that the scores for each question add up to the correct total given by `\questiontotal`. (Default.)

notesallowed Specifies that notes are allowed in the exam.

nonotesallowed Specifies that notes are not allowed in the exam. (Default.)

noscorecheck Don't check the scores.

10pt Set the normal font size to 10pt.

11pt Set the normal font size to 11pt.

12pt Set the normal font size to 12pt. (Default.)

14pt Set the normal font size to 14pt.

17pt Set the normal font size to 17pt.

20pt Set the normal font size to 20pt.

a4paper Set the paper size to A4. (Default.)

a3paper Set the paper size to A3.

oneside Set the page formatting for one-sided printing.

twoside Set the page formatting for double-sided printing. (Default.)

draft Draft mode on and puts “DRAFT” banner across every page. This also sets draft mode for all loaded packages, included `graphicx`, so you won't see included images. You can override this with

```
\setkeys{Gin}{draft=false}
```

final Draft mode off. (Default.)

leqno Left equation numbers.

fleqn Flush left equations.

`\ueaexamhook`

```
\ueaexamhook
```

If this command has been defined before `ueaexam` is loaded, it will be executed during the option processing stage. This can be used to set up a script to generate exam sheet and solution sheet without having to modify the document. For example, suppose the document is called `exam2011.tex` and loads `ueaexam` without any class options:

```
\documentclass{ueaexam}
```

then the exam sheet (`exam2011.pdf`) can be create using:

```
pdflatex exam2011
```

and the solution sheet (`exam2011-solutions.pdf`) can be created using (no line breaks):

```
pdflatex -jobname exam2011-solutions  
"\def\ueaexamhook{\showanswerstrue}\input{exam2011}"
```

4 Large Font Sizes for Partially Sighted Students

As from version 2.01, there are extra class options `14pt`, `17pt` and `20pt` in order to produce larger font sizes. Note that to use these options you must have the `extsizes` bundle installed².

Returning to the example in the previous section where the document is called `exam2011.tex`, the large-font exam sheet (`exam2011-large.pdf`) can be created using (no line breaks):

```
pdflatex -jobname exam2011-large  
"\def\ueaexamhook{\setptsize{20}\setexampapersize{a3}}"  
\input{exam2011}
```

Alternatively, you can use an external application to resize the A4 document. For example (no line breaks):

```
gs -q -dNOPAUSE -dBATCH -sDEVICE=pdfwrite -sPAPERSIZE=a3  
-dFIXEDMEDIA -dPDFFitPage -sOutputFile=exam2011-large.pdf  
exam2011.pdf
```

²If you don't have this package it can be download from <http://www.ctan.org/pkg/extsizes>

5 Available Commands

5.1 Preamble Commands

The following commands may be used in the preamble:

`\questiontotal` `\questiontotal{<number>}`

Used to specify the total number of marks for each question. (Default: 15). If the marks for a question don't add up to this value, a warning is issued in draft mode and an error is issued in final mode. You can suppress the score check using the `noscorecheck` class option.

`\gquestiontotal` `\gquestiontotal{<number>}`

As `\questiontotal` but has a global effect.

`\university` `\university{<university name>}`

Used to specify the name of the university. Defaults to "University of East Anglia".

`\school` `\school{<school name>}`

Used to specify the name of the school. Defaults to "School of Computing Sciences".

`\sitting` `\sitting{<text>}`

Used to specify the sitting of the exam. For example:

```
\sitting{May/June UG}
```

or

```
\sitting{UG Reassessment/Delayed First Sit}
```

`\examyyear` `\examyyear{<year>}`

Used to specify the academic year (e.g. 2010/11). Defaults to the current academic year.

`\course` `\course{<code>}{<name>}`

Used to specify the course code and name. For example:

```
\course{CMP-1A4Y}{Programming --- Languages and Software Construction}
```

`\timeallowed` `\timeallowed{<time limit>}`

Used to specify the time allowed. For example:

```
\timeallowed{2 hours}
```

`\version` `\version{<number>}`

Used to specify the exam version number, as required by the new regulations. For example:

```
\version{1}
```

`\contact` `\contact{<name>}`

Used to specify the module contact name, as required by the new regulations. For example:

```
\contact{Dr A.N. Other, CMP}
```

`\rubric` `\rubric{<text>}`

The rubric on the front page regarding the number of questions to be done in each section can be automatically generated using the `\section` command (see below), however additional information can be added using the command `\rubric`. For example:

```
\rubric{Use separate answer books for each section.}
```

`\turnovertrue` `\turnovertrue`

This will put the words “TURN OVER” at the bottom of each page³ except for the last page. (This is the default.)

`\turnoverfalse` `\turnoverfalse`

Prevents the words “TURN OVER” appearing at the bottom of each page.

5.2 Document Commands

The following commands may be used within the document:

`\maketitle` `\maketitle`

³odd pages only, if two sided printing specified

This makes the title page. It should be the first command in the document environment.

`\score` `\score{<number>}`

This is used to indicate the maximum number of marks for a question or part of question. For example:

Find the derivative of $f(x)=x^2 - x + 1$.`\score{5}`

This adds *<number>* to the running total for the current question and displays the mark using

`\scoreformat` `\scoreformat{<number>}`

which has the default definition: `\marginpar{[\marklabel{<mark>}]}` where

`\marklabel` `\marklabel{<mark>}`

prints *<mark>* followed by either “mark” (where *<mark>* is 1) or “marks”.

The `\score` command must be used within an enumerate environment.

Warning: don't attempt to use or redefine `\mark` or `\marks` to specify the mark. These commands are TeX primitives and should not be meddled with.

`\section` `\section{<Number of Questions to Be Answered>}`

This is the only sectioning command defined within this class file. This command issues a `\clearpage`, prints “SECTION” followed by the section letter (e.g. A), and on the following line it prints the argument. Both lines are centrally aligned. As usual, the section can be referenced using `\label` and `\ref`. For example, the input:

```
\section{2}
```

would produce the output:

SECTION A
Answer TWO questions from this Section.

The `\section` command may occur within an enumerate or any of the other list-making environments, provided at least one `\item` precedes it. This means that all the exam questions can be placed within a single enumerate environment, ensuring consistent numbering throughout the document. Note that there is no starred version of this command.

If the argument \langle *Number of Questions to Be Answered* \rangle is a number or the strings `all` or `the`, the full section title and rubric information will be generated automatically. For example:

```
\section{all}
```

will produce:

SECTION A
Answer ALL questions from this Section.

and it will add “ALL questions from Section A” to the rubric information on answering questions.

Alternatively, the section title and the rubric information can be explicitly entered using:

```
\section[rubric info]{title text}
```

For example:

```
\section[ANY question]{Answer ANY question.}
```

Note: It is best not to have any commands within the optional argument of `\section`, unless they expand to a simple text string^a. At best, this will cause L^AT_EX to keep complaining that the title page is not up to date, at worse it will cause a “T_EX capacity exceeded” error.

^aThis shouldn't be much of a problem as it's unlikely that there will be any, although it does mean that you can't use any spacing macros (such as `~`) either.

```
\addtorubric \addtorubric{text}
```

This may be used to insert any additional text to the rubric. The above note also applies to this command.

```
\Solution \Solution{text}
```

(Note the initial capital letter.) This may be used to specify the solution to the problem. The solution is only displayed if the `solutions` class option is used. Figures and tables within the argument of `\Solution` will have different numbering to those outside of `\Solution`. This ensures that the figures and tables that form part of the questions retain the same numbering in the solution sheet. The solution text is formatted according to

```
\solutionfont \solutionfont
```


This defaults to slanted sans-serif.

As from v4.08, the argument of `\Solution` may contain verbatim text.

6 Defining Problems and their Solutions in External Files

It's possible to define the exam questions in a different file. This makes it easier to change the ordering of the questions within the exam. Questions can be defined using:

`defproblem`

```
\begin{defproblem}{\label}  
<text>  
\end{defproblem}
```

For example:

```
\begin{defproblem}{compute-factorial}  
  Compute  $4!$ ,  $5!$  and  $6!$ .\score{3}  
  \Solution  
  {%  
    \begin{align*}  
      4! &= 4 \times 3 \times 2 \times 1 = 24\\  
      5! &= 5 \times 4! = 5 \times 24 = 120\\  
      6! &= 6 \times 5! = 6 \times 120 = 720  
    \end{align*}  
  }  
\end{defproblem}
```

There is a short-cut command

`\newproblem`

```
\newproblem{\label}{\question}{\answer}
```

For example, the above can also be written as:

```
\newproblem{compute-factorial}  
{%  
  Compute  $4!$ ,  $5!$  and  $6!$ .\score{3}  
}  
{%  
  \begin{align*}  
    4! &= 4 \times 3 \times 2 \times 1 = 24\\  
    5! &= 5 \times 4! = 5 \times 24 = 120\\  
    6! &= 6 \times 5! = 6 \times 120 = 720  
  \end{align*}  
}
```

`textenum`

```
\begin{textenum}
```

You can use the `textenum` environment for in-line numbered lists. This environment uses the same counter as the corresponding `enumerate` level, so answers that require longer passages than the question can use the same numbering system. For example:

```
\newproblem{compute-factorial}
{%
  Compute
  \begin{textenum}
  \item $4!$, \item $5!$ and \item $6!$.
  \end{textenum}\score{3}
}
{%
  \begin{enumerate}
  \item $4! = 4 \times 3 \times 2 \times 1 = 24$
  \item $5! = 5 \times 4! = 5 \times 24 = 120$
  \item $6! = 6 \times 5! = 6 \times 120 = 720$
  \end{enumerate}
}
```

In your document, you can load the questions using one of the following:

`\loadallproblems` `\loadallproblems [<db name>] {<file>}`

Loads all problems defined in *<file>* into a database named *<db name>*.

`\loadselectedproblems` `\loadselectedproblems [<db name>] {<label list>} {<file>}`

Loads the problems defined in *<file>* whose labels are given in *<label list>* into a database named *<db name>*.

`\loadexceptproblems` `\loadexceptproblems [<db name>] {<label list>} {<file>}`

Loads the problems defined in *<file>* whose labels are not given in *<label list>* into a database named *<db name>*.

`\loadrandomproblems` `\loadrandomproblems [<db name>] {<n>} {<file>}`

Loads *<n>* randomly selected problems defined in *<file>* into a database named *<db name>*.

Once you have loaded the required problems, you can either explicitly select problems using:

`\useproblem` `\useproblem [<db name>] {<label>}`

or you can iterate over all problems using:

`\foreachproblem` `\foreachproblem [<db name>] {<body>}`

Within *(body)*, you can use

`\thisproblem`

```
\thisproblem
```

to use the current problem and

`\thisproblemlabel`

```
\thisproblemlabel
```

to access the current label.

7 Example Documents

The following is a short sample document illustrating the use of this class file:

```
\documentclass{ueaexam}

\course{ABC-1XY}{SAMPLE COURSE}
\timeallowed{2 hours}

\version{1}
\contact{Dr A.N. Other}

\begin{document}
\maketitle
\section{the}
\begin{enumerate}

\item This is the first question, it has two parts.

\begin{enumerate}
\item The first part \score{20}
\Solution{This is the solution to the first part.}
\item The second part \score{20}
\Solution{This is the solution to the second part.}
\end{enumerate}

\section{2}

\item This is the first question of the second part,
but because we are still in the same enumerate environment,
this question is question number 2. If the last line of this
paragraph is long, it will run into the marks so in this
case, we can put the marks on the following line to make it neater.
\par\mbox{ }\score{30}
\Solution{This is the solution.}

\item This is question number 3.\score{30}
\Solution{This is the solution to question number 3.}
```

```

\item This is the last question.\score{30}
\Solution{This is the solution to the last question.}
\end{enumerate}
\end{document}

```

In the following example, the exam is made up of three sections, where each section is written by a different lecturer (call them Dr A, Dr B and Dr C.) Rather than the lecturers trying to determine who has the most up-to-date version of the file, the questions for each section are defined in three separate files, say A.tex, B.tex and C.tex. In this way, each lecturer can independently edit their own questions.

Suppose Dr A is providing questions on counting, then A.tex might look like:

```

\newproblem{compute-factorial}
{%
  Compute
  \begin{textenum}
  \item $4!$, \item $5!$ and \item $6!$.
  \end{textenum}\score{15}
}
{%
  \begin{enumerate}
  \item $4! = 4 \times 3 \times 2 \times 1 = 24$
  \item $5! = 5 \times 4! = 5 \times 24 = 120$
  \item $$6! = 6 \times 5! = 6 \times 120 = 720$
  \end{enumerate}
}

\newproblem{factorial-terms}%
{%
  Write in terms of factorials:
  \begin{textenum}
  \item $21 \times 20$,
  \item $\frac{1}{9 \times 8}$,
  \item $42$.
  \end{textenum}\score{15}
}%
{%
  \begin{enumerate}
  \item $21 \times 20 = \frac{21!}{19!}$
  \item $\frac{1}{9 \times 8} = \frac{7!}{9!}$
  \item $42 = \frac{42!}{41!}$
  \end{enumerate}
}

```

Suppose Dr B is providing questions on differentiation, then B.tex might look like:

```

\begin{defproblem}{diff-f}
  Differentiate each of the following functions with respect to $x$:
  \begin{enumerate}
  \item $f(x) = x^2$.\score{5}

```

```

\Solution{$f'(x) = 2x$.}
\item $f(x) = 3x^3$.\score{5}
\Solution{$f'(x) = 9x^2$.}
\item $f(x) = 2x^2 + x$.\score{5}
\Solution{$f'(x) = 4x + 1$.}
\end{enumerate}
\end{defproblem}

```

Suppose Dr C is providing questions on set theory, then C.tex might look like:

```

\newproblem{sets-showequal}%
{%
  Which of these sets are equal:
  ${a, b, c}$,
  ${c, b, a}$,
  ${c, b, b, a}$,
  ${a, c, b, c}$?\score{15}
}%
{%
  They are all equal. Order and repetition do not change a set.
}

```

The main file might then look as follows:

```

\documentclass{ueaexam}

\course{ABC-2XY}{SAMPLE COURSE II}
\timeallowed{3 hours}
\rubric{Use a separate answer book for each section.}
\version{1}
\contact{Dr A. Other, CMP}

\loadallproblems[counting]{A}
\loadallproblems[differentiation]{B}
\loadallproblems[sets]{C}

\begin{document}
\maketitle

\section{2}
\begin{enumerate}
\foreachproblem[counting]{\item\thisproblem}

\section{1}

\foreachproblem[differentiation]{\item\thisproblem}

\section{all}

\foreachproblem[sets]{\item\thisproblem}

```

```
\end{enumerate}  
\end{document}
```

Acknowledgements

Some of the code was amended by G. Janacek.

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the definition.

A		enumerate package 2	P	
\addtorubric 8	environments:	defproblem 9	package options:	noscorecheck 5
amsmath package 2	defproblem 9	enumerate <u>2</u> , <u>7</u> , <u>10</u>	paralist package 2	probsoln package 2
article class 1	enumerate <u>2</u> , <u>7</u> , <u>10</u>	inparaenum <u>2</u>		
article.cls class 1	inparaenum <u>2</u>	textenum <u>2</u> , <u>9</u> , <u>10</u>		
C		etoolbox package 2	Q	
calc package 2	\examyyear 5	\examsizes package 2	\questiontotal <u>3</u> , <u>5</u>	
class options:	exsizes package 2	extsizes package 4		
10pt <u>3</u>			R	
11pt <u>3</u>	F		\rubric 6	
12pt <u>3</u>	fntcount package 2		S	
14pt <u>2-4</u>	\foreachproblem 10		\school 5	
17pt <u>2-4</u>	fp package 2		\score 7	
20pt <u>2-4</u>			\scoreformat 7	
a3paper <u>3</u>	G		\section <u>6</u> , <u>7</u>	
a4paper <u>3</u>	geometry package 2		\sitting 5	
draft <u>3</u>	\questiontotal 5		\Solution <u>3</u> , <u>3</u> , <u>8</u>	
final <u>4</u>	graphicx package <u>2</u> , <u>3</u>		\solutionfont 8	
fleqn <u>4</u>			sys-exam package 1	
leqno <u>4</u>	H			
nonotesallowed <u>3</u>	helvet package 2		T	
noscorecheck <u>3</u>			textenum (environment)	
nosolutions <u>3</u>	I	 <u>2</u> , <u>9</u> , <u>10</u>	
notesallowed <u>3</u>	ifthen package 2		\thisproblem 11	
notimes <u>3</u>	inparaenum (environ-		\thisproblemlabel . . . 11	
oneside <u>3</u>	ment) 2		\timeallowed 6	
scorecheck <u>3</u>	\item <u>7</u>		\turnoverfalse 6	
solutions <u>3</u> , <u>8</u>			\turnovertrue 6	
times <u>2</u> , <u>3</u>	L			
twoside <u>3</u>	lmodern package 2		U	
watermarkfirst <u>3</u>	\loadallproblems 10		ueaexam class 1, 4	
\contact 6	\loadexceptproblems . 10		\ueaexamhook 4	
courier package 2	\loadrandomproblems . 10		\university 5	
\course 5	\loadselectedproblems		\useproblem 10	
 10			
D		M		
defproblem (environ-	\maketitle 6		V	
ment) 9	\marklabel 7		\version 6	
drawwatermark package . . 2	mathptmx package 2			
E		N		
enumerate (environ-	\newproblem 9		X	
ment) <u>2</u> , <u>7</u> , <u>10</u>			xcolor package 2	