OU-SUPPS

Class files for Open University teaching material

Robert Brignall*

Version 1.32
2016-08-30

Contents

1 Introduction

The OU-SUPPS repository contains four class files:

- `ouab.cls` for Assignment Booklets and TMA booklets;
- `outn.cls` for Tutor Notes and specimen solutions (for distribution to students);
- `ouexam.cls` for exams, specimen exams, and solutions;

*For bug reports, comments and suggestions go to `https://github.com/rbrignall/OU-SUPPS`.

- `ouicma.cls` for print versions of iCMAs (e.g. for offender learners).

In addition, template LaTeX files for each class file are available, illustrating most of the features available.

2 Installation

OU-SUPPS does not need installation. To use:

1. Download the contents of the OU-SUPPS repository from `https://github.com/rbrignall/OU-SUPPS`.
2. Save the contents in your working folder.
3. Open the template tex file for the class you want to use with your LaTeX editor, and compile it *twice* (using either `latex` or `pdflatex`).
4. Your LaTeX distribution may ask to install a few packages the first time you compile each of the template files.
5. If it compiled successfully, then everything is working as expected.

Important: When writing a SUPP file using the new system, as a minimum you will need to have the following files in your working folder:

- The tex source for your document.
- The relevant class file (e.g. `ouab.cls` if you're creating an Assignment Booklet).
- The two OU logo files.

3 Using `ouab.cls`

This section covers the features available in `ouab.cls`, for writing Assignment Booklets and TMA books. `ouab.cls` is based on the standard `article` class file, so any commands used there can be used in `ouab`.

3.1 Preamble

3.1.1 Options

As well as the default version, there are two options available. These options do not affect the syntax you use for the rest of your file.

`\documentclass{ouab}`

This is the default operation for using `ouab.cls`. It produces a table of contents on the front page, and is designed for Assignment Booklets containing more than one TMA/CMA.

`\documentclass[oneassignment]{ouab}`

This version is designed for booklets that contain (e.g.) a single TMA (although it will let you have more than one assignment in the source). Instead of a table of contents on the front page, it inserts information provided by the first assignment in the `latex` file.

`\documentclass[markcheck]{ouab}`

(EXPERIMENTAL) This version can be used to check that the stated total marks for a question (i.e. the number n in `\question[...]{n}[...]`) is equal to the number of marks produced by `\marks{...}` in the subquestions. If the numbers do not add up, it interrupts the compilation and reports this. Do not wholly rely on this, as it is an experimental feature!

`\documentclass[oneassignment,markcheck]{ouab}`

As always, you can combine options and the effects of each option will be seen.

3.1.2 Commands to be used in the preamble

`\faculty{...}`

The name of the faculty (optional, default is 'Faculty of Science, Technology, Engineering and Mathematics')

`\modulecode{...}`

Sets the code of the module (required).

`\moduletitle{...}`

Sets the title (i.e./ name) of the module (required).

`\abtitle{...}`

Sets the title for the Assignment Booklet (required).

`\absubtitle{...}`

Subtitle for the Assignment Booklet (optional, default is blank).

`\abyear{...}`

Sets the year/presentation for this AB (required).

`\copyrightyear{...}`

Sets the date for copyright, used in the footer on the front page (optional, default is same as `\abyear`).

`\suppno{...}`

Sets the SUPP number, mainly used by DPU/LTS (optional, default is 'DRAFT').

`\versionno{...}`

Sets the version number, mainly used by DPU/LTS (optional, default is blank).

`\optiontext{...}`

Changes the text that appears at the top of multi-choice questions (optional, default is 'Options:'). Does not need to be in the preamble, so you can change the text for options part-way through the document.

`\nocutoffdate`

This switches off the text "Cut-off date" that appears (e.g.) on the front page (optional). Use this for modules where the cut-off dates are not provided in the assignment booklet.

`\abinstructions{...}`

Command for the instructions/rubric on the front page of the booklet (optional, but defaults to 'No special instructions specified.')

3.2 Creating assignments

Three commands are available to create assignments: TMA, CMA, and a generic command for creating any other type.

`\tma[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`
`\tma*[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`

Creates a TMA with number equal to ⟨*number*⟩. The ⟨*date*⟩ and ⟨*subtitle*⟩ parts are both optional and can be omitted. The starred version suppresses the `\newpage` command, so does not insert a new page before the assignment.

⟨*date*⟩ specifies the cut-off date for the TMA.

⟨*subtitle*⟩ specifies text for the 'subtitle', which is only used in the description of the assignment on the front page.

`\cma[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`
`\cma*[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`

Creates a CMA, usage as per `\tma`. The starred version suppresses the `\newpage` command, so does not insert a new page before the assignment.

`\assignment{`⟨*name*⟩`}[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`
`\samepageassignment{`⟨*name*⟩`}[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`

Creates an assignment type ⟨*name*⟩ (required), numbered with ⟨*number*⟩. The ⟨*date*⟩ and ⟨*subtitle*⟩ parts are both optional and can be omitted. The `\samepageassignment` variant creates an assignment without inserting a new page, whereas `\assignment` inserts a new page.

All assignment types add an entry to the table of contents on the front page, and reset the `question` counter to 0. The behaviour of the table of contents depends on whether the `oneassignment` option has been specified or not.

3.3 Creating questions

The internal counter used for generating questions is `question`. This gets reset at the start of each assignment.

3.3.1 Question syntax

`\question[`⟨*description*⟩`]{`⟨*marks*⟩`}`

Creates a new question, with the number of marks available specified by the command ⟨*marks*⟩.

The parameter ⟨*description*⟩ is optional, and can be used to provide additional information about the question in its header line.

`\question*[`⟨*description*⟩`]`

Starred variant of `\question` which does not require the number of marks to be specified. The parameter ⟨*description*⟩ is optional.

3.3.2 Subquestions

To create subquestions, use the standard LaTeX `enumerate` environment.

```
\begin{enumerate}
\item ...
\item ...
...
\end{enumerate}
```

These environments can be nested to create subsubquestions, etc. The default numbering style is `(a),(b),...` for subquestions, and `(i),(ii),...` for subsubquestions.

The class file `ouab.cls` loads the `enumitem` package, which provides a number of features, two of which we list here.

- Changing the numbering: Start the environment using, e.g. `\begin{enumerate}[A.]`, to modify the numbering system to `A.,B.,...`.
- Resuming numbering after a break: use `\begin{enumerate}[resume]` to continue counting from the previous time an enumerate environment at this level was called.

For fuller details of the possibilities with `enumitem`, see the documentation on its CTAN entry.

3.3.3 Marks

Marks can be placed anywhere in the document (including inside displayed equations). You may need to compile your `tex` file twice in order for the marks to be correctly aligned.

`\marks{`⟨*n*⟩`}`

Places ⟨*n*⟩ in square brackets in the right hand margin on the line.

Warning: This feature has redefined the TeX primitive `\marks` command.

`\mk{`⟨*n*⟩`}`

A synonym for `\marks`, for those moving from OUTeX.

If using the experimental `markcheck` option, when compiled, LaTeX will check whether the sum of the entries inside `\marks` and `\mk` commands adds up to the parameter ⟨*marks*⟩ specified by the preceding `\question`. This has some limitations (e.g. it won't understand `\tfrac{1}{2}`).

If the numbers are not the same, `markcheck` will cause the compilation to stop and give you information about where the marks do not add up. Press ⟨*enter*⟩ to carry on compiling your tex file.

3.3.4 Multiple choice

For CMAs and other assignment types that have multiple choice questions, the following commands are available.

The text appearing at the top of each list of options can be changed using the `\optiontext{`⟨*title*⟩`}` command. This command can be used in the premable, or at any point in the document.

```
\begin{options}
   \item ...
   \item ...
\end{options}
```

Creates a list of options, with the text of each `\item` starting on a new line. Options are labelled `A.`,`B.`,`C.`,....

```
\begin{inlineoptions}[⟨line spacing⟩]{⟨columns⟩}
   \item ...
   \item ...
\end{inlineoptions}
```

Creates a list of options, displayed in ⟨*columns*⟩ number of columns, equally spaced. Each successive `\item` is placed sequentially from left to right, then starting a new line as necessary. Options are labelled `A.`,`B.`,`C.`,....

The optional parameter ⟨*line spacing*⟩ can be used to increase the spacing between each line. However, note that this also affect spacing within options, which can have undesired effect if your options include, e.g., a `matrix` environment.

```
\noitem
```

This command can be used within `inlineoptions` to leave a 'blank' in the list of options. This is particularly useful on the final line if you have fewer `\item`s left than the specified number of ⟨*columns*⟩.

3.4 Technical info

This section can be omitted unless you need/want to know a bit more about the class file.

3.4.1 Packages loaded

`ouab.cls` loads the following packages automatically, so you do not need to call `\usepackage{...}` to use the features provided by these.

- `geometry` to set page margins, sizes, etc
- `fancyhdr` for formatting headings
- `graphicx` for handling images, etc
- `changepage` to adjust page widths automatically
- `array` extends the `array` and `tabular` environments.
- `amsmath` loaded with the `fleqn` option to left-align displayed mathematics.
- `amssymb` extra mathematical symbols
- `lastpage` for knowing how many pages the document contains
- `ifthen` for internal latex coding
- `enumitem` with options `inline` and `shortlabels`. This is a powerful tool that will help preparing subquestions (see later).
- `marginnote` for handling marks, etc in the margin.
- `caption` with option `labelsep=quad` to improve formatting for figures.

4 Using outn.cls

This section covers the features available in `outn.cls`, for writing Tutor Notes and Specimen Solutions for assignments. `outn.cls` is based on the standard `article` class file, so any commands used there can be used in `outn`. Note also that the syntax is as close as possible to that created in `ouab.cls`, so, in theory, the two types of SUPP could be generated from a common source at a reasonably advanced stage of development.

4.1 Preamble

4.1.1 Options

As well as the default version to produce tutor notes, there are two options available. These options do not affect the syntax you use for the rest of your file.

`\documentclass{outn}`

This is the default operation for using `outn.cls`. It produces a Tutor Notes booklet.

`\documentclass[specsolns]{outn}`

This version is designed to create a specimen solutions PDF. In this mode, a number of commands have no effect, e.g. those for specifying marks (`\marks` and `\subtotal`, etc), the title is changed to reflect the different type of document, and (by default) the font colour is changed to green.

`\documentclass[studenttex]{outn}`

(EXPERIMENTAL) The idea is that on compilation, as well as producing Tutor Notes (or specimen solutions if `specsolns` is also specified) it produces a compilable `latex` source file called ⟨*filename*⟩`-student.tex` that can be compiled by (e.g.) ALs who do not have the `OU-SUPPS` suite of class files on their machine. This system is still under development, and is subject to some limitations in functionality.

4.1.2 Commands to be used in the preamble

`\faculty{...}`

The name of the faculty (optional, default is 'Faculty of Science, Technology, Engineering and Mathematics')

`\modulecode{...}`

Sets the code of the module (required).

`\moduletitle{...}`

Sets the title (i.e./ name) of the module (required).

`\tntitle{...}`

Sets the title for the Tutor Notes/Specimen Solutions (required).

`\tnyear{...}`

Sets the year/presentation for this TN (required).

`\copyrightyear{...}`

Sets the date for copyright, used in the footer on the front page (optional, default is same as `\tnyear`).

`\specsolnscolor{`⟨*color*⟩`}`

Command to change the font color used in Specimen Solutions to ⟨*color*⟩ (e.g. `blue` or `red`). The class file loads the package `xcolor.sty` so you can use this to specify your own colors. This command is optional, the default color is `ougreen`, which is defined in `outn.cls` by `\definecolor{ougreen}{RGB}{0,128,0}`.

`\optiontext{...}`

Changes the text that appears at the top of multi-choice questions (optional, default is 'Options:'). Does not need to be in the preamble, so you can change the text for options part-way through the document.

4.2 Creating assignments

Three commands are available to create assignments: TMA, CMA, and a generic command for creating any other type. These commands have the same syntax as in `ouab`, but the behaviour is slightly different (since `outn` does not produce a table of contents).

`\tma[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`
`\tma*[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`

Creates a TMA with number equal to ⟨*number*⟩. The ⟨*date*⟩ and ⟨*subtitle*⟩ parts are both optional and can be omitted. The starred version suppresses the `\newpage` command, so does not insert a new page before the assignment.

⟨*date*⟩ specifies the cut-off date for the TMA.

⟨*subtitle*⟩ has no effect in `outn`. It is parsed by the system and then ignored, so that the syntax is identical to that used by `ouab`.

`\cma[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`
`\cma*[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`

Creates a CMA, usage as per `\tma`. The starred version suppresses the `\newpage` command, so does not insert a new page before the assignment.

`\assignment{`⟨*name*⟩`}[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`
`\samepageassignment{`⟨*name*⟩`}[`⟨*date*⟩`]{`⟨*number*⟩`}[`⟨*subtitle*⟩`]`

Creates an assignment type ⟨*name*⟩ (required), numbered with ⟨*number*⟩. The ⟨*date*⟩ and ⟨*subtitle*⟩ parts are both optional and can be omitted (and ⟨*subtitle*⟩ has no effect in `outn`). The `\samepageassignment` variant creates an assignment without inserting a new page, whereas `\assignment` inserts a new page.

All assignment types reset the `question` counter to 0.

4.3 Creating questions/solutions

The internal counter used for generating questions/solutions is `question`. This gets reset at the start of each assignment.

4.3.1 Question syntax

`\question[`⟨*description*⟩`]{`⟨*marks*⟩`}`

Creates a new question (or solution), with the number of marks available specified by the command ⟨*marks*⟩. The wording at the top of the question is 'Solution to Question ⟨*n*⟩'.

The parameter ⟨*description*⟩ is optional, and can be used to provide additional information about the question in its header line.

`\question*[`⟨*description*⟩`]`

Starred variant of `\question` which does not require the number of marks to be specified. The parameter ⟨*description*⟩ is optional.

`\begin{solution}...\end{solution}`

An environment to contain the solution to a question. The contents of these environments are precisely what is output to the `latex` source file when using the experimental `studenttex` mode. If not using `studenttex` mode, this environment is merely a semantic feature, and does not need to be used.

4.3.2 Subquestions

To create subquestions, use the standard LaTeX `enumerate` environment.

```
\begin{enumerate}
\item ...
\item ...
...
\end{enumerate}
```

These environments can be nested to create subsubquestions, etc. The default numbering style is `(a),(b),...` for subquestions, and `(i),(ii),...` for subsubquestions.

The class file `outn.cls` loads the `enumitem` package, which provides a number of features, two of which we list here.

- Changing the numbering: Start the environment using, e.g. `\begin{enumerate}[A.]`, to modify the numbering system to `A.,B.,...`.
- Resuming numbering after a break: use `\begin{enumerate}[resume]` to continue counting from the previous time an enumerate environment at this level was called.

For fuller details of the possibilities with `enumitem`, see the documentation on its CTAN entry.

4.3.3 Marks

Marks can be placed anywhere in the document (including inside displayed equations). You may need to compile your `tex` file twice in order for the marks to be correctly aligned. In `specsolns` mode, all the commands in this subsubsection are suppressed in the resulting PDF. In `studenttex` mode, all these commands are suppressed in the outputted `latex` source file.

`\marks{`⟨*n*⟩`}`

Places ⟨*n*⟩ in the right hand margin on the line.

Warning: This feature has redefined the TeX primitive `\marks` command.

`\mk{`⟨*n*⟩`}`

A synonym for `\marks`, for those moving from OUTeX.

`\solnmarksplus{`⟨*description*⟩`}{`⟨*n*⟩`}`

Place ⟨*n*⟩ in the right hand margin, and ⟨*description*⟩ (which could be a short sentence running to several lines) to the right of the mark.

`\subtotal[`⟨*description*⟩`]{`⟨*n*⟩`}`
`\total[`⟨*description*⟩`]{`⟨*n*⟩`}`

Specifies that the subtotal or total for a part-question or question is ⟨*n*⟩, placing ⟨*n*⟩ in the margin, and the word 'Subtotal' or 'Total' in large, bold to its right. The optional argument ⟨*description*⟩ can be used to append wording to the word 'Total'. Typical usages of the ⟨*description*⟩ parameter are:
`\subtotal[for part \theenumi]{`⟨*n*⟩`]}`
`\total[for Question \thequestion]{`⟨*n*⟩`]}`.

`\subtotal*[`⟨*description*⟩`]`
`\total*[`⟨*description*⟩`]`

Calculates the subtotal or total for a part-question or question, placing the calculated value in the margin, and the word 'Subtotal' or 'Total' in large, bold to its right; the calculated value accounts for all marks given in any of the following commands: `bracketedmarks`, `solnmarks`, `solnmarksplus`, `marks`, `mk`, and `mkplus`. The optional argument ⟨*description*⟩ is the same as with the `\subtotal` and `\total` commands.

`\setSubtotalHline[`⟨*options*⟩`]`

Sets options (including turning off) for horizontal lines after `\subtotal` and `\subtotal*` command. Supported options, in any order, are:

- `moveleft=`⟨*length*⟩, default is `.1\textwidth`: specifies the horizontal offset of the line;
- `width=`⟨*length*⟩, default is `1.5\textwidth`: specifies the horizontal width of the line;
- `height=`⟨*length*⟩, default is `3pt`: specifies the height of the line;
- `color=`⟨*color*⟩, default is `black`: specifies the colour of the line;
- `draw line=`⟨*true/false*⟩, default is `true`: specifies if the line is to be drawn or not.

Simply calling `\setSubtotalHline` without options, resets all of the options to their default values.

Sample uses include, for example:

```
\setSubtotalHline[moveleft=0pt]
```

```
\setSubtotalHline[moveleft=0pt,height=5pt,color=orange]
```

This command is just a short cut to the `\pgfkeys` command, and users familiar with the syntax of `\pgfkeys` might prefer to use, for example:

```
\pgfkeys{/subtotalhline,default,moveleft=-3cm,color=red!40!white}
```

`\setTotalHline[`⟨*options*⟩`]`

This is analogous to the `\setSubtotalHline` command, and customises the lines drawn (if any) after the `\total` and `\total*` commands. It takes exactly the same options as `\setSubtotalHline`, and they perform the analogous tasks.

Users who prefer to use \pgfkeys directly may use, for example,

```
\pgfkeys{/totalhline,default,moveleft=-3cm,color=red!40!white}
```

4.3.4 Multiple choice

For CMAs and other assignment types that have multiple choice questions, the following commands are available. It is unlikely that these will be needed in tutornotes, but they have been included for consistency with ouab.cls.

The text appearing at the top of each list of options can be changed using the \optiontext{⟨*title*⟩} command. This command can be used in the premable, or at any point in the document.

```
\begin{options}
   \item ...
   \item ...
\end{options}
```

Creates a list of options, with the text of each \item starting on a new line. Options are labelled A.,B.,C.,....

```
\begin{inlineoptions}[⟨line spacing⟩]{⟨columns⟩}
   \item ...
   \item ...
\end{inlineoptions}
```

Creates a list of options, displayed in ⟨*columns*⟩ number of columns, equally spaced. Each successive \item is placed sequentially from left to right, then starting a new line as necessary. Options are labelled A.,B.,C.,....

The optional parameter ⟨*line spacing*⟩ can be used to increase the spacing between each line. However, note that this also affect spacing within options, which can have undesired effect if your options include, e.g., a matrix environment.

```
\noitem
```

This command can be used within inlineoptions to leave a 'blank' in the list of options. This is particularly useful on the final line if you have fewer \items left than the specified number of ⟨*columns*⟩.

4.4 Comments for tutors or students

Since the same latex source is expected to produce both tutor notes and (using specsolns mode) specimen solutions, commands are provided for comments to appear to tutors and/or students.

Warning: These commands are still under development and syntax/appearance is still liable to change in future.

```
\textref{⟨description⟩}
```

Inserts ⟨*description*⟩ in bold and italics, in a right-aligned box. Primarily expected to be used for references to texts or handbooks.

```
\remark{⟨text⟩} \begin{longremark}⟨text⟩\end{longremark}
```

Provides a command and environment in which to write text that is visible only in the

tutornotes (i.e. it is suppressed in `specsolns` mode). The `longremark` environment is designed for longer paragraphs of explanation.

Warning: Do not use the `longremark` *environment inside the* `solutions` *environment! In certain modes this will cause errors relating to verbatim mode.*

4.5 Technical info

This section can be omitted unless you need/want to know a bit more about the class file.

4.5.1 Packages loaded

`outn.cls` loads the following packages automatically, so you do not need to call `\usepackage{...}` to use the features provided by these.

- `geometry` to set page margins, sizes, etc
- `fancyhdr` for formatting headings
- `graphicx` for handling images, etc
- `changepage` to adjust page widths automatically
- `array` extends the `array` and `tabular` environments.
- `amsmath` loaded with the `fleqn` option to left-align displayed mathematics.
- `amssymb` extra mathematical symbols
- `lastpage` for knowing how many pages the document contains
- `ifthen` for internal latex coding
- `enumitem` with options `inline` and `shortlabels`. This is a powerful tool that will help preparing subquestions (see later).
- `marginnote` for handling marks, etc in the margin.
- `caption` with option `labelsep=quad` to improve formatting for figures.

5 Using `ouexam.cls`

This section covers the features available in `ouexam.cls`, for writing exam papers, specimen (and second specimen) papers, and solutions to exams.

In theory, the same `latex` source file can be used to generate both an exam script and its model solutions, which might be helpful for drafting purposes. For the final version to be handed over, it is advised that you create two source files (one for the exam, one for its solutions).

5.1 Preamble

5.1.1 Options

As well as the default version, there are three options available. These options do not affect the syntax you use for the rest of your file, except for some of the commands in the preamble (detailed later).

`\documentclass{ouexam}`

This is the default operation for using `ouexam.cls` for setting exam papers. Any text in `solution` environments is ignored.

`\documentclass[specimen]{ouexam}`

This version is designed for specimen exam papers. The formatting on the front page is changed, and certain commands (e.g. `\examtime`) are not required.

`\documentclass[secondspecimen]{ouexam}`

Version for second specimen exam papers. Formatting on the front page and requirements are similar to the `specimen` option.

`\documentclass[showsolutions]{ouexam}`

This version creates a solution booklet. Formatting on the front page is changed, and the contents of `solution` environments gets displayed.

```
\documentclass[specimensolutions]{ouexam}
\documentclass[specimen,showsolutions]{ouexam}
```

The `showsolutions` option can be used alongside the `specimen` option to provide model solutions for specimen exams formatted in a style expected by LTS. The major changes in this mode are the formatting of teh title, footer information, and text within the `solutions` environment will be displayed without additional formatting. The `specimensolutions` option is an alias for `specimen,showsolutions`.

`\documentclass[14pt]{ouexam}`

This provides a large-print version for visually impaired students. Also available are `17pt` and `20pt`, and these all work with other options. These options should only be used in response to specific requests made by students, and should not be used to produce the standard exam.

5.1.2 Commands to be used in the preamble

`\modulecode{...}`

Sets the code of the module (required).

`\session{...}`

Sets the session code for the exam (required, except for `specimen` and `secondspecimen`). This replaces the deprecated `\conflation{...}` command (although this still works to ensure backwards compatibility).

`\moduletitle{...}`

Sets the title (i.e./ name) of the module (required).

`\specimensolutionstitle{...}`

Optional. Sets the title that appears at the top of the specimen solutions in `specimensolutions` mode. Default is “Solutions to the second examination paper”. It has no function in other modes.

`\examcode{`⟨*code*⟩`}`

Encodes ⟨*code*⟩ for the barcode on the front page (optional, default is blank).

`\examtime{`⟨*start*⟩ `--` ⟨*finish*⟩`}`
`\examday{`⟨*day*⟩`}`
`\exammonth{`⟨*month*⟩`}`
`\examyear{`⟨*year*⟩`}`

Specifies the time, day, month and year of the exam (required, except for `specimen` and `secondspecimen`).

`\timeallowed{`⟨*n*⟩ `hours}`

Specifies the time allowed (required).

`\copyrightyear{...}`

Sets the date for copyright, used in the footer on the front page (optional, default is same as `\examyear`).

`\suppno{...}`

Sets the SUPP number for `specimen` and `secondspecimen` options, mainly used by DPU/LTS (optional, default is blank).

`\versionno{...}`

Sets the version number for `specimen` and `secondspecimen` options, mainly used by DPU/LTS (optional, default is blank).

`\faculty{...}`

Sets the name for the faculty (appears in `specimensolutions` mode only). The default is 'Faculty of Science, Technology, Engineering and mathematics'.

`\optiontext{...}`

Changes the text that appears at the top of multi-choice questions (optional, default is 'Options:'). Does not need to be in the preamble, so you can change the text for options part-way through the document.

`\examinstructions{...}`

Command for the instructions/rubric on the front page of the exam (optional, but defaults to 'No special instructions specified.')

5.2 Creating questions

The internal counter used for generating questions is `question`.

5.2.1 Question syntax

`\question[`⟨*description*⟩`]`

Creates a question. The parameter ⟨*description*⟩ is optional, and can be used to provide

additional information about the question in its header line.

5.2.2 Subquestions

To create subquestions, use the standard LaTeX `enumerate` environment.

```
\begin{enumerate}
\item ...
\item ...
...
\end{enumerate}
```

These environments can be nested to create subsubquestions, etc. The default numbering style is `(a),(b),...` for subquestions, and `(i),(ii),...` for subsubquestions.

The class file `ouexam.cls` loads the `enumitem` package, which provides a number of features, two of which we list here.

- Changing the numbering: Start the environment using, e.g. `\begin{enumerate}[A.]`, to modify the numbering system to `A.,B.,...`.
- Resuming numbering after a break: use `\begin{enumerate}[resume]` to continue counting from the previous time an enumerate environment at this level was called.

For fuller details of the possibilities with `enumitem`, see the documentation on its CTAN entry.

5.2.3 Marks

Marks can be placed anywhere in the document (including inside displayed equations). You may need to compile your `tex` file twice in order for the marks to be correctly aligned.

`\marks{⟨n⟩}`

Places ⟨n⟩ in square brackets in the right hand margin on the line.

Warning: This feature has redefined the TeX primitive `\marks` command.

`\mk{⟨n⟩}`

A synonym for `\marks`, for those moving from OUTeX.

5.2.4 Multiple choice

For exams that have multiple choice questions, the following commands are available. The syntax is the same as for `ouab.cls` and `ouicma.cls`.

The text appearing at the top of each list of options can be changed using the `\optiontext{⟨title⟩}` command. This command can be used in the premable, or at any point in the document.

```
\begin{options}
   \item ...
   \item ...
\end{options}
```

Creates a list of options, with the text of each `\item` starting on a new line. Options are labelled `A.,B.,C.,...`.

```
\begin{inlineoptions}[⟨line spacing⟩]{⟨columns⟩}
   \item ...
   \item ...
\end{inlineoptions}
```

Creates a list of options, displayed in ⟨*columns*⟩ number of columns, equally spaced. Each successive `\item` is placed sequentially from left to right, then starting a new line as necessary. Options are labelled `A.`,`B.`,`C.`,....

The optional parameter ⟨*line spacing*⟩ can be used to increase the spacing between each line. However, note that this also affect spacing within options, which can have undesired effect if your options include, e.g., a `matrix` environment.

```
\noitem
```

This command can be used within `inlineoptions` to leave a 'blank' in the list of options. This is particularly useful on the final line if you have fewer `\item`s left than the specified number of ⟨*columns*⟩.

5.3 Creating solutions

```
\begin{solution}
...
\end{solution}
```

Enter the contents of solutions inside the `solution` environment. This is simply a wrapper environment, and can be used within an `enumerate` environment (i.e. solutions per subquestion/subsubquestion), and/or you can use `enumerate` environments inside the solution environment.

The text inside the `solution` environment is *only* displayed when one of the `showsolutions` or the `specimensolutions` options is specified in the preamble.

When displayed, the solutions can be formatted in one of two ways:

`[specimensolutions]` **or** `[specimen,showsolutions]`**:** The text is displayed without further formatting.

`[showsolutions]`**:** The text of the solutions is set to `\small`, and placed on a beige background with a thick black line on the left, to indicate the scope of the solution.

Note: The showing/hiding solutions features of `ouexam` are designed for use by authors when drafting exams, to help you check answers tally up to the corresponding questions. When creating the final version for handover, you will probably want to make two copies of your working file, one to create the exam (with solutions removed), the other to create the solutions (possibly with the questions removed, though this is option except for specimen solutions).

5.3.1 Specifying marks within the `solution` environment

```
\marks{⟨n⟩}
\mk{⟨n⟩}
```

These two commands can be used to specify marks, as per the format for creating questions. Within the `solution` environment in `[showsolutions]` mode, the numbers are displayed in the right-hand margin in text size `\small`, and without square brackets.

```
\solnmarksplus{⟨n⟩}{⟨comment⟩}
\mkplus{⟨n⟩}{⟨comment⟩}
```

These two commands can be used to enter ⟨n⟩ marks in the right hand column (as per `\marks`), but additionally enters ⟨*comment*⟩ in the margin, to the right of ⟨n⟩.

5.4 Alternative method to create solutions

If you prefer not to have solutions appearing in-line with the questions, an alternative method is to place all solutions at the end. You should not use this method for creating specimen solutions.

`\solutions`

This flag behaves similarly to the `\appendix` flag in standard LaTeX files: here, `\solutions` marks the end of the exam paper, resets the `question` counter, and redefines the text in the `\question` command to 'Solution to Question `\thequestion`'.

Solutions can then be created using the `\question` command to create questions, and all other commands are available.

Important note: Contents after the `\solutions` flag will be displayed irrespective of whether the `showsolutions` option is specified in the preamble. However, you may want to specify `showsolutions` in order to adjust the formatting of the cover page.

5.5 Technical info

This section can be omitted unless you need/want to know a bit more about the class file.

5.5.1 Packages loaded

`ouexam.cls` loads the following packages automatically, so you do not need to call `\usepackage{...}` to use the features provided by these.

- `geometry` to set page margins, sizes, etc
- `fancyhdr` for formatting headings
- `graphicx` for handling images, etc
- `changepage` to adjust page widths automatically
- `array` extends the `array` and `tabular` environments.
- `amsmath` loaded with the `fleqn` option to left-align displayed mathematics.
- `amssymb` extra mathematical symbols
- `lastpage` for knowing how many pages the document contains
- `ifthen` for internal latex coding
- `enumitem`. This is a powerful tool that will help preparing subquestions (see later).
- `marginnote` for handling marks, etc in the margin.
- `mdframed` with option `xcolor` to provide formatting for coloured boxes to display solutions.
- `verbatim` to ensure the contents of the `solution` environment is hidden unless `showsolutions` is specified.
- `barcodes` for creating the barcode on the front page.
- `extsizes` when options `14pt`, `17pt` or `20pt` are used, to handle larger font sizes.

6 Using `ouicma.cls`

Not yet written!

7 Moving from OUTeX

Some features in OUTeX have not been carried over. Here is some help to help you adjust to standard LaTeX.

OUTeX	LaTeX

Figures	Use the standard LaTeX figure environment. e.g.
	`\begin{figure}\label{`⟨*label*⟩`}`
	`\centering`
	`\includegraphics{`⟨*image*⟩`}`
	`\caption{`⟨*caption*⟩`}`
	`\end{figure}`
	See sources online for options to float figures, etc.

Compiling: `latex`, `dvips` + `ps2pdf` with `eps` files	You can use `pdflatex` with `eps` files, but to do so you need to add this to the preamble: `\usepackage{epstopdf}` The alternative is to use the old route `latex`, `dvips` + `ps2pdf`.
`\intertext`	`\end{enumerate}...\begin{enumerate}[resume]`

`\<...&...\>`	`\begin{align*}...&...\end{align*}`
	You can recover the OUTeX command by inserting the following into your preamble:
	`\def\<#1\>{\begin{align*}#1\end{align*}}`
	However, we do not recommend you use this, as it will not be understood by anyone not previously familiar with OUTeX (e.g. ALs), and will mean snippets of your code won't work out of the context of the file it's in.

`\[ ... \\ ... \]`	`\beglin{gather*}...\\...\end{gather*}`

Matrix alignment	Load the `mathtools` package, so you can write e.g.
	`\begin{pmatrix}[l]`
	⟨*matrix*⟩
	`\end{pmatrix}`
	to left-align all columns. Alternatively (and to specify different column alignments) use the `array` environment instead, e.g.:
	`\left(\begin{array}{lrcl}`
	⟨*matrix*⟩
	`\end{array}\right)`.

Binomials (column vector format)	`\binom{n}{k}`
`\setright{\text{`⟨*text*⟩`}}`	`\tag{`⟨*text*⟩`}`