

# The automultiplechoice package\*

Alexis Bienvenue  
paamc@passoire.fr

December 6, 2023

## Abstract

This package helps designing multiple choice exams ready for automated marking from papers scans.

Answers and questions are optionally shuffled, creating different sheets for every student.

## 1 Introduction

The package `automultiplechoice` helps formatting multiple choice questionnaires with automated marking from papers scans in mind:

- The package can produce different copies of the question sheet for each student, optionally shuffling answers and questions for each student.
- Markers can be printed on each sheet, so as to be able to analyse scans after examination. All the needed information about the position of the markers and the boxes to be checked by the students is given in an auxiliary file during  $\text{\LaTeX}$  run.

See Auto Multiple Choice (AMC) software (<https://www.auto-multiple-choice.net/>) for an integration of this package, with user interface for automated marking.

## 2 Samples

We begin with several samples to see what can be done with the `automultiplechoice` package. All `automultiplechoice` commands and options will be detailed further.

For all these samples, two sets of questions are used: a group of geography questions, and a group of history questions. These are defined in a common  $\text{\LaTeX}$  file named `questions.tex`:

```
\element{geography}{  
  \begin{question}{Ghana}  
    What is the capital of Ghana?  
    \begin{choiceshoriz}  
      \correctchoice{Accra}  
      \wrongchoice{Addis Abeba}  
      \wrongchoice{Ankara}  
      \wrongchoice{Apia}
```

---

\*This document corresponds to version revision: `r:8721bfba` from AMC 1.6.0+git20231206122656

```

        \end{choiceshoriz}
    \end{question}
}

\element{geography}{
    \begin{question}{Thailand}
        What is the capital of Thailand?
        \begin{choiceshoriz}
            \correctchoice{Bangkok}
            \wrongchoice{Banjul}
            \wrongchoice{Beijing}
            \wrongchoice{Beirut}
            \wrongchoice{Berlin}
        \end{choiceshoriz}
    \end{question}
}

\element{geography}{
    \begin{question}{Egypt}
        What is the capital of Egypt?
        \begin{choices}
            \correctchoice{Cairo}
            \wrongchoice{Caracas}
            \wrongchoice{Cayenne}
            \wrongchoice{Chisinau}
            \wrongchoice{Conakry}
        \end{choices}
    \end{question}
}

\element{geography}{
    \begin{question}{Ireland}
        What is the capital of Ireland?
        \begin{multicols}{3}
            \begin{choices}
                \correctchoice{Dublin}
                \wrongchoice{Dili}
                \wrongchoice{Djibouti}
                \wrongchoice{Doha}
                \wrongchoice{Dakar}
                \wrongchoice{Dhaka}
            \end{choices}
        \end{multicols}
    \end{question}
}

\element{history}{
    \begin{questionmult}{1901}
        Which of the following events are taking place during the year

```

```

1901?
\begin{choices}
  \correctchoice{Funeral of Queen Victoria in London}
  \correctchoice{Official end of the Caste War of Yucat\'an}
  \wrongchoice{King George of Greece becomes absolute monarch of Crete}
  \wrongchoice{The first line of the Paris M\'etro is opened}
\end{choices}
\end{questionmult}
}

\element{history}{
  \begin{questionmult}{1850}
    Which of the following events are taking place during the year
    1850?
    \begin{choices}
      \correctchoice{American Express is founded by Henry Wells \& William Fargo}
      \wrongchoice{Napoleon Bonaparte crosses the Alps and invades Italy}
      \wrongchoice{Kwang-su becomes emperor of China}
      \wrongchoice{First horse-drawn omnibuses established in London}
    \end{choices}
  \end{questionmult}
}

\element{history}{
  \begin{questionmult}{1971}
    Which of the following events are taking place during the year
    1971?
    \begin{choices}
      \correctchoice{Apollo 14 lands on the Moon}
      \correctchoice{The Soviet Union launches Salyut 1}
      \correctchoice{Death of Louis Armstrong}
      \wrongchoice{The first commercial Concorde flight takes off}
    \end{choices}
  \end{questionmult}
}

```

We will ask automultiplechoice package to include two geography questions and two history questions at random for each student, shuffling questions and answers, with the following code:

```

\cleargroup{all}
\shufflegroup{geography}
\copygroup[2]{geography}{all}
\shufflegroup{history}
\copygroup[2]{history}{all}
\shufflegroup{all}
\insertgroup{all}

```

You can read these commands as “clear group **all**, shuffle questions inside group **geography** and copy the first two to group **all**, do the same for group **history**, shuffle the four questions copied into **all** and print them”.

## 2.1 Standard layout

A set of 30 students sheets can be produced from the following L<sup>A</sup>T<sub>E</sub>X source named `sample-amc.tex`:

```
\documentclass{article}
\usepackage{automultiplechoice}
\usepackage{multicol}
\begin{document}

\input{questions.tex}

\onecopy{30}{

\noindent{\bf AMC \hfill SAMPLE TEST}

\vspace{3ex}
```

For this test, package `{\sf automultiplechoice}` is used without any option. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `{\tt nowatermark}` option.

Commands from `{\sf automultiplechoice}` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

```
\vspace{3ex}

\cleargroup{all}

\shufflegroup{geography}
\copygroup[2]{geography}{all}
\shufflegroup{history}
\copygroup[2]{history}{all}
\shufflegroup{all}
\insertgroup{all}

}
\end{document}
```

producing a 30-pages document (every page has number 1), from which we show the first pages on page 7.

Note that “DRAFT” indications can be cancelled using option `nowatermark`, or using AMC software.

You can see on each page markers that can be used for automated completed answer sheets scans analysis:

- Four circles ● are printed in the corners, to be able to analyse any rotation or scaling of the scans.
- Binary boxes are printed in the header area, so as to be able to read student sheet number and page number. On page 2 for example, you can see that these binary boxes are coding 2/1/59:



+2/1/59+

Here, 2 is the student sheet number, 1 is the page number for this student, and 59 is a checking value that can be used for checking correct identification from a scan.

If you also use `calibration` option, `automultiplechoice` will produce a `.xy` file with informations about the exact position in the page of all the markers, and all the boxes. This option is automatically set by AMC software, which then use the information in the `.xy` file for automated marking.

## 2.2 Separate answer sheet

In some situations, you may need a separate answer sheet:

- this makes cheating even more difficult;
- this can reduce the number of pages to scan.

This is done using `separateanswersheet` option of `automultiplechoice` package. You also have to use commands `\AMCformBegin` to indicate the beginning of this separate answer sheet (usually after a `\clearpage` or `\AMCcleardoublepage` command), and `\AMCform` to insert the form to be completed by the students, as in the following example (`sample-separate.tex`):

```
\documentclass{article}
\usepackage[separateanswersheet]{automultiplechoice}
\usepackage{multicol}
\begin{document}
```

```
\input{questions.tex}
```

```
\onecopy{30}{
```

```
\noindent{\bf AMC \hfill SAMPLE TEST}
```

```
\vspace{3ex}
```

For this test, package `{\sf automultiplechoice}` is used with `{\tt separateanswersheet}` option, so that all answers are to be filled on a separate sheet by students. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `{\tt nowatermark}` option.

Commands from `{\sf automultiplechoice}` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

```
\vspace{3ex}
```

```
\cleargroup{all}
```

```

\shufflegroup{geography}
\copygroup[2]{geography}{all}
\shufflegroup{history}
\copygroup[2]{history}{all}
\shufflegroup{all}
\insertgroup{all}

```

```

\clearpage

```

```

\AMCformBegin

```

This is the answer sheet: all answers are to be ticked on this page to be taken into account.

```

\vspace{2ex}

```

```

\AMCform

```

```

}
\end{document}

```

First pages of the result are shown on page 8. There are now 2 pages per student: the first with questions, and the second for answers. Only the second will be completed by the students, and scanned for analysis.

## 2.3 Without markers

With the `nopage` option, package `automultiplechoice` does not include any page markers for scan processing. I'm afraid you can't use any automated marking software with this layout, but you can still use answer sheet and corrected answer sheet (option `indivanswers`, added here) for a manual marking...

The L<sup>A</sup>T<sub>E</sub>X source `sample-plain.tex` that only differs from `sample-amc.tex` by its options passed to `automultiplechoice`:

```

\usepackage[nopage,indivanswers]{automultiplechoice}

```

produces a 30-pages document, from which we show the first pages on page 9.

# First pages from L<sup>A</sup>T<sub>E</sub>X source detailed in section 2.1 – see sample-amc.pdf

●  ● +1/1/60+

AMC SAMPLE TEST

For this test, package `automultiplechoice` is used without any option. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `\nocancelmark` option.

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☐ The Soviet Union launches Salyut 1  
☐ The first commercial Concorde flight takes off  
☐ Death of Louis Armstrong  
☐ Apollo 14 lands on the Moon

**Question 2** What is the capital of Egypt?

☐ Cayenne  
☐ Caracas  
☐ Cuito  
☐ Conakry  
☐ Chisinau

**Question 3** Which of the following events are taking place during the year 1850?

☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ First horse-drawn omnibuses established in London  
☐ American Express is founded by Henry Wells & William Fargo  
☐ Kwang-on becomes emperor of China

**Question 4** What is the capital of Ghana?

☐ Accra ☐ Addis Ababa ☐ Ankara ☐ Apia

● For your examination, preferably print documents compiled from `automultiple-choice`. ●

●  ● +2/1/60+

AMC SAMPLE TEST

For this test, package `automultiplechoice` is used without any option. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `\nocancelmark` option.

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1901?

☐ The first line of the Paris Metro is opened  
☐ Official end of the Caste War of Yucatán  
☐ King George of Greece becomes absolute monarch of Creta  
☐ Funeral of Queen Victoria in London

**Question 2** What is the capital of Ireland?

☐ Djibouti ☐ Dhaka ☐ Dhaka  
☐ Dili ☐ Dili ☐ Dakar

**Question 3** What is the capital of Ghana?

☐ Apia ☐ Accra ☐ Addis Ababa ☐ Ankara

**Question 4** Which of the following events are taking place during the year 1850?

☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ First horse-drawn omnibuses established in London  
☐ American Express is founded by Henry Wells & William Fargo  
☐ Kwang-on becomes emperor of China

● For your examination, preferably print documents compiled from `automultiple-choice`. ●

●  ● +3/1/52+

AMC SAMPLE TEST

For this test, package `automultiplechoice` is used without any option. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `\nocancelmark` option.

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☐ The Soviet Union launches Salyut 1  
☐ The first commercial Concorde flight takes off  
☐ Apollo 14 lands on the Moon  
☐ The Soviet Union launches Salyut 1  
☐ Death of Louis Armstrong

**Question 2** Which of the following events are taking place during the year 1850?

☐ First horse-drawn omnibuses established in London  
☐ Kwang-on becomes emperor of China  
☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ American Express is founded by Henry Wells & William Fargo

**Question 3** What is the capital of Ireland?

☐ Dhaka ☐ Dhaka ☐ Dakar  
☐ Dili ☐ Dili ☐ Djibouti

**Question 4** What is the capital of Thailand?

☐ Beijing ☐ Bangkok ☐ Bangkok ☐ Beirut ☐ Berlin

● For your examination, preferably print documents compiled from `automultiple-choice`. ●

●  ● +4/1/52+

AMC SAMPLE TEST

For this test, package `automultiplechoice` is used without any option. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `\nocancelmark` option.

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☐ The Soviet Union launches Salyut 1  
☐ Apollo 14 lands on the Moon  
☐ Death of Louis Armstrong  
☐ The first commercial Concorde flight takes off

**Question 2** What is the capital of Egypt?

☐ Caracas  
☐ Cayenne  
☐ Cuito  
☐ Conakry  
☐ Chisinau

**Question 3** Which of the following events are taking place during the year 1850?

☐ American Express is founded by Henry Wells & William Fargo  
☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ First horse-drawn omnibuses established in London  
☐ Kwang-on becomes emperor of China

**Question 4** What is the capital of Ireland?

☐ Djibouti ☐ Dhaka ☐ Dakar  
☐ Dili ☐ Dili ☐ Dublin

● For your examination, preferably print documents compiled from `automultiple-choice`. ●

First pages from L<sup>A</sup>T<sub>E</sub>X source detailed in section 2.2 – see sample-separate.pdf

•  • +1/1/60+

AMC SAMPLE TEST

For this test, package `automultiplechoice` is used with `separateanswersheet` option, so that all answers are to be filled on a separate sheet by students. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `nowatermark` option.

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☐ A The Soviet Union launches Salyut 1  
☐ B The first commercial Concorde flight takes off  
☐ C Death of Louis Armstrong  
☐ D Apollo 14 lands on the Moon

**Question 2** What is the capital of Egypt?

☐ A Cayenne  
☐ B Caracas  
☐ C Cairo  
☐ D Cusack  
☐ E Chisum

**Question 3** Which of the following events are taking place during the year 1850?

☐ A Napoleon Bonaparte crosses the Alps and invades Italy  
☐ B First horse-drawn omnibuses established in London  
☐ C American Express is founded by Henry Wells & William Fargo  
☐ D Kwang-on becomes emperor of China

**Question 4** What is the capital of Ghana?

☐ A Accra ☐ B Addis Ababa ☐ C Ankara ☐ D Apia

• For your examination, preferably print documents compiled from `automultiple-choice`. •

•  • +1/2/60+

This is the answer sheet: all answers are to be ticked on this page to be taken into account.

**Question 1:** ☐ A ☐ B ☐ C ☐ D  
**Question 2:** ☐ A ☐ B ☐ C ☐ D  
**Question 3:** ☐ A ☐ B ☐ C ☐ D  
**Question 4:** ☐ A ☐ B ☐ C ☐ D

• For your examination, preferably print documents compiled from `automultiple-choice`. •

•  • +2/1/60+

AMC SAMPLE TEST

For this test, package `automultiplechoice` is used with `separateanswersheet` option, so that all answers are to be filled on a separate sheet by students. Page markers are printed in view of an automated marking from papers scans. DRAFT indications can be cancelled using `nowatermark` option.

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1901?

☐ A The first line of the Paris Metro is opened  
☐ B Official end of the Cato War of Yunán  
☐ C King George of Greece becomes absolute monarch of Crete  
☐ D Funeral of Queen Victoria in London

**Question 2** What is the capital of Ireland?

☐ A Djibouti ☐ B Dhaka ☐ C Doha  
☐ D Dublin ☐ E Dili ☐ F Dakar

**Question 3** What is the capital of Ghana?

☐ A Apia ☐ B Accra ☐ C Addis Ababa ☐ D Ankara

**Question 4** Which of the following events are taking place during the year 1850?

☐ A Napoleon Bonaparte crosses the Alps and invades Italy  
☐ B First horse-drawn omnibuses established in London  
☐ C American Express is founded by Henry Wells & William Fargo  
☐ D Kwang-on becomes emperor of China

• For your examination, preferably print documents compiled from `automultiple-choice`. •

•  • +2/2/60+

This is the answer sheet: all answers are to be ticked on this page to be taken into account.

**Question 1:** ☐ A ☐ B ☐ C ☐ D  
**Question 2:** ☐ A ☐ B ☐ C ☐ D  
**Question 3:** ☐ A ☐ B ☐ C ☐ D  
**Question 4:** ☐ A ☐ B ☐ C ☐ D

• For your examination, preferably print documents compiled from `automultiple-choice`. •



# First pages from L<sup>A</sup>T<sub>E</sub>X source detailed in section 2.3 – see sample-plain.pdf

AMC

SAMPLE TEST

For this test, package `automultiplechoice` is used with the following options:

- `nopage`, so that no page markers are printed: nothing is planned for future automated marking from papers scans.
- `indivanswers`, so that correct answers are indicated (this is the corrected answer sheet. Without this option, you get the question sheet).

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☒ The Soviet Union launches Soyuz 1  
☐ The first commercial Concorde flight takes off  
☒ Death of Louis Armstrong  
☒ Apollo 14 lands on the Moon

**Question 2** What is the capital of Egypt?

☐ Cayenne  
☐ Caracas  
☒ Cairo  
☐ Cusco  
☐ Chisinau

**Question 3** Which of the following events are taking place during the year 1850?

☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ First horse-drawn omnibuses established in London  
☒ American Express is founded by Henry Wells & William Fargo  
☐ Kwang-sun becomes emperor of China

**Question 4** What is the capital of Ghana?

☒ Accra  
☐ Addis Ababa  
☐ Ankara  
☐ Apia

1

AMC

SAMPLE TEST

For this test, package `automultiplechoice` is used with the following options:

- `nopage`, so that no page markers are printed: nothing is planned for future automated marking from papers scans.
- `indivanswers`, so that correct answers are indicated (this is the corrected answer sheet. Without this option, you get the question sheet).

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1901?

☐ The first line of the Paris Metro is opened  
☒ Official end of the Cretan War of Vassilata  
☐ King George of Greece becomes absolute monarch of Crete  
☒ Funeral of Queen Victoria in London

**Question 2** What is the capital of Iceland?

☐ Djibouti  
☐ Dhaka  
☒ Doha  
☐ Dili  
☐ Dakar

**Question 3** What is the capital of Ghana?

☐ Apia  
☒ Accra  
☐ Addis Ababa  
☐ Ankara

**Question 4** Which of the following events are taking place during the year 1850?

☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ First horse-drawn omnibuses established in London  
☒ American Express is founded by Henry Wells & William Fargo  
☐ Kwang-sun becomes emperor of China

1

AMC

SAMPLE TEST

For this test, package `automultiplechoice` is used with the following options:

- `nopage`, so that no page markers are printed: nothing is planned for future automated marking from papers scans.
- `indivanswers`, so that correct answers are indicated (this is the corrected answer sheet. Without this option, you get the question sheet).

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☐ The first commercial Concorde flight takes off  
☒ Apollo 14 lands on the Moon  
☒ The Soviet Union launches Soyuz 1  
☒ Death of Louis Armstrong

**Question 2** Which of the following events are taking place during the year 1850?

☐ First horse-drawn omnibuses established in London  
☐ Kwang-sun becomes emperor of China  
☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☒ American Express is founded by Henry Wells & William Fargo

**Question 3** What is the capital of Iceland?

☐ Dhaka  
☐ Dili  
☐ Doha  
☒ Djibouti  
☐ Dhaka

**Question 4** What is the capital of Thailand?

☐ Beijing  
☐ Banjul  
☒ Bangkok  
☐ Beirut  
☐ Berlin

1

AMC

SAMPLE TEST

For this test, package `automultiplechoice` is used with the following options:

- `nopage`, so that no page markers are printed: nothing is planned for future automated marking from papers scans.
- `indivanswers`, so that correct answers are indicated (this is the corrected answer sheet. Without this option, you get the question sheet).

Commands from `automultiplechoice` are used to print, for each student, two geography questions and two history questions, at random. Questions and answers are shuffled.

**Question 1** Which of the following events are taking place during the year 1971?

☒ The Soviet Union launches Soyuz 1  
☒ Apollo 14 lands on the Moon  
☒ Death of Louis Armstrong  
☐ The first commercial Concorde flight takes off

**Question 2** What is the capital of Egypt?

☐ Caracas  
☐ Cayenne  
☒ Cairo  
☐ Cusco  
☐ Chisinau

**Question 3** Which of the following events are taking place during the year 1850?

☒ American Express is founded by Henry Wells & William Fargo  
☐ Napoleon Bonaparte crosses the Alps and invades Italy  
☐ First horse-drawn omnibuses established in London  
☐ Kwang-sun becomes emperor of China

**Question 4** What is the capital of Iceland?

☐ Djibouti  
☐ Dili  
☐ Doha  
☒ Dhaka  
☐ Dakar

1

## 3 Usage

### 3.1 Package options

The following options are available for package `automultiplechoice`:

`noshuffle` cancels answers shuffling for all questions.

`noshufflegroups` cancels groups shuffling.

`answers` produces a common corrected answers sheet.

`indivanswers` shows the boxes that corresponds to correct choices on the question sheet.

`box` includes every question in a  $\text{\LaTeX}$  box, so that they can't be cutted on two different pages.

`asbox` does the same for questions in the separate answer sheet.

`separateanswersheet` asks for a separate answer sheet (see section 2.2 for an example). Commands `\AMCformBegin` and `\AMCform` must be used to describe the separate answer sheet (see section 3.6).

`digits` puts digits instead of letters in the boxes, when `separateanswersheet` (or `insidebox`) is used.

`outsidebox` prints boxes labels outside the boxes on the answersheet when `separateanswersheet` is set.

`init` initializes the random generator from time. *This option is only for testing: don't use it for a real exam!*

`completemulti` adds an answer "None of these answers are correct." at the end of each multiple question (question with no, one or several correct answers), so as to make the difference between "I don't know" and "I think none of the answers are correct".

`insidebox` puts a letter (or a digit if `digits` option is used) inside the boxes, even if `separateanswersheet` is not used. The `insidebox` option is implicitly called when using `separateanswersheet`: no need to call it then.

`calibration` asks for logging positions of boxes and markers in the `.xy` file. Without this option, a  $\text{\LaTeX}$  run updates the document but not the `.xy` file.

`nowatermark` cancels the "DRAFT" indications above pages.

`catalog` is used for formatting a catalog of questions, not an exam. Then the question identifiers will be printed.

`keys` defines the way the question identifiers will be printed on the catalog file. With `keys=next` (the default), the question identifiers will be printed next to the questions numbers. With `keys=line`, the question identifiers will be printed on one line before the question text, so that the question will look close to the final result on the exam copies.

`francais` asks for french localisation.

`lang=XX` asks for localisation in XX language. At present, only CA (Catalan), DE (German), ES (Spanish), FR (French), IT (Italian), JA (Japanese), NO (Norwegian) and NL (Dutch) are available.

`plain` cancels `environ` and `etex` automatic loading. The default behaviour is to load `environ` and `etex` packages if available, as they improve `automultiplechoice`. This is not done when `plain` option is set.

`nopage` cancels markers print and page layout definition (see sample in section 2.3).

`automarks` , when used with `separateanswersheet`, cancels markers print on the subject page (they are only shown on the answer sheet pages).

`postcorrect` tells that correct answers won't be given in the LaTeX source. The teacher will fill one answer sheet for AMC to analyse the scan and set correct answers from it.

`fullgroups` cancels the use of the optional parameter of `\insertgroup` and `\copygroup`, so that each group is always fully inserted and fully copied.

`storebox` asks to use `\storebox` instead of `\savebox` to handle ovals (when using oval shape). The package `storebox` will be loaded.

`pdfform` use this option to produce PDF forms. The PDF sheet won't be printed, but filled by each student with a PDF reader. The completed PDF will then be sent to the teacher, and given to AMC for data capture.

See also section 3.8 for a french version of some of these options.

## 3.2 Questions and answers

We make a difference between two kind of multiple choice questions:

- **Simple questions:** there is one and only one correct choices among the proposed choices, *and this is announced to the student*. Thus, the student is asked to check one answer if he thinks this is the good one, and to check none if he has no idea.
- **Multiple questions:** there can be zero, one or several correct choices among the proposed choices. This is also announced to the student (using the `\multiSymbole` sign, with default ♣), so that the student is asked to check all the boxes corresponding to correct choices, and to let unchecked all boxes corresponding to wrong choices.

`question (env.)` Simple questions are enclosed in a `{question}{<id>}` environment, and multiple questions `questionmult (env.)` are enclosed in a `{questionmult}{<id>}` environment. These environments contain the question text, and the proposed choices inside a `choices`-like environment (see next). The `<id>` argument is a question identifier. Each question must have a unique identifier, different from the other questions identifiers.

---

Questions environments

---

**Question 1** What is the elevation of Mount Everest?

- ☐ 8,253 m  
☐ 8,810 m  
☐ 8,848 m

**Question 2 ♣** Which contries are in the Americas?

- ☐ Cambodia  
☐ Guatemala  
☐ Canada  
☐ Switzerland

```
\begin{question}{everest}
  What is the elevation of
  Mount Everest?
  \begin{choices}
    \correctchoice{8,848\,m}
    \wrongchoice{8,253\,m}
    \wrongchoice{8,810\,m}
  \end{choices}
\end{question}

\begin{questionmult}{americas}
  Which contries are in the Americas?
  \begin{choices}
    \correctchoice{Guatemala}
    \correctchoice{Canada}
    \wrongchoice{Switzerland}
    \wrongchoice{Cambodia}
  \end{choices}
\end{questionmult}
```

`\AMCcompleteMulti` For multiple questions, it is sometimes useful to make the difference between a student who thinks that none of he choices are correct, and a student who did not answer the question. `\AMCnoCompleteMulti` The use of package option `completemulti` can be used in this case: it adds a choice to all multiple questions. Commands `\AMCcompleteMulti` and `\AMCnoCompleteMulti` can also be used to change this behaviour for a single question.

---

Additional answer “none” for a single question

---

**Question 3 ♣** Which contries are in the Americas?

- ☐ Guatemala  
☐ Cambodia  
☐ Canada  
☐ Switzerland  
☐ *None of these answers are correct.*

```
\begin{questionmult}{americas}
  \AMCcompleteMulti
  Which contries are in the Americas?
  \begin{choices}
    \correctchoice{Guatemala}
    \correctchoice{Canada}
    \wrongchoice{Switzerland}
    \wrongchoice{Cambodia}
  \end{choices}
\end{questionmult}
```

`choices` (*env.*)  
`choiceshoriz` (*env.*)  
`choicescustom` (*env.*)

Depending on the formatting style for answers, one can choose one of the following ones:

- Environment `choices` is usually chosen for long answers:

The choices environment

**Question 4 ♣** What are the possible uses of latex?

- ☐ Latex is used as a fuel for some space launch vehicles.
- ☐ Latex from the chicle and jelutong trees is used in chewing gum.
- ☐ Natural rubber is the most important product obtained from latex.

```
\begin{questionmult}{latex}
What are the possible uses of latex?
\begin{choices}
\correctchoice{Natural rubber is
the most important product
obtained from latex.}
\correctchoice{Latex from the chicle
and jelutong trees is used in
chewing gum.}
\wrongchoice{Latex is used as a fuel
for some space launch vehicles.}
\end{choices}
\end{questionmult}
```

- environment `choiceshoriz` is chosen for short answers:

The choiceshoriz environment

**Question 5** How many legs for an insect?

- ☐ 2      ☐ 6      ☐ 8

```
\begin{question}{insect}
How many legs for an insect?
\begin{choiceshoriz}
\correctchoice{6}
\wrongchoice{2}
\wrongchoice{8}
\end{choiceshoriz}
\end{question}
```

- environment `choicescustom` is provided to customize answers formatting. See 3.9.3 for details.

`\correctchoice`      As you have seen in these examples, the `choices`-like environments contain `\correctchoice{<text>}` and `\wrongchoice` and `\wrongchoice{<text>}` commands, with the text of the proposed choice as argument.

### 3.3 Scoring

`\scoring` Scoring strategies can be given in the L<sup>A</sup>T<sub>E</sub>X source. They don't have any impact on the question sheet: they are only transmitted to the analysis software through the `.amc` file. See AMC `\scoringDefaultM` documentation to write proper commands for your needs. `\scoring{<score>}` can be used inside a `\question` or `\questionmult` environment to describe the scoring strategy for the question, or after a `\correctchoice` or `\wrongchoice` command to describe score associated to a particular choice. `\scoringDefaultM{<score>}` and `\scoringDefaultS{<score>}` define default scoring strategies for multiple and simple questions. `\QuestionIndicative` tags a question that is not taken into account to compute the mark – for example, it can be used for a question about the way students have enjoyed the course.

### 3.4 Groups of questions

Several commands are available that allows shuffling questions for each question sheet. They handle groups of questions. These groups will usually contain questions, but can be made of any L<sup>A</sup>T<sub>E</sub>X content.

**\element** The command `\element{⟨groupname⟩}{⟨content⟩}` adds element with content `⟨content⟩` to the group named `⟨groupname⟩`. The command `\shufflegroup{⟨groupname⟩}` shuffles elements of group named `⟨groupname⟩`. The command `\insertgroup[⟨n⟩]{⟨groupname⟩}` inserts elements of group `⟨groupname⟩` one after one. If optional parameter `⟨n⟩` is given, only the first `⟨n⟩` elements of the group are inserted in the document. If not, or if `⟨n⟩` is negative, all the elements are inserted. The command `\insertgroupfrom[⟨n⟩]{⟨groupname⟩}{⟨i⟩}` does the same, starting from element at index `⟨i⟩` (the first element has index 0).

As an example without questions in groups elements, let us create a small group named **serie**, containing five elements, and play with it:

Managing groups	
Numbers: one two three four five. Three numbers from the second (index=1) one: two three four. Two of them after shuffling: two four.	<pre> \element{serie}{ one} \element{serie}{ two} \element{serie}{ three} \element{serie}{ four} \element{serie}{ five} Numbers:\insertgroup{serie}.  Three numbers from the second (index=1) one:\insertgroupfrom[3]{serie}{1}.  \shufflegroup{serie} Two of them after shuffling:\insertgroup[2]{serie}. </pre>

**\cleargroup** The command `\cleargroup{⟨groupname⟩}` clears all the elements of group `⟨groupname⟩`, making an empty group. The command `\copygroup[⟨n⟩]{⟨from⟩}{⟨to⟩}` copies the elements of group `⟨from⟩` to group `⟨to⟩` – if optional parameter `⟨n⟩` is given, only the `⟨n⟩` first elements are copied. If not, or if `⟨n⟩` is negative, all the elements are copied. The command `\copygroupfrom[⟨n⟩]{⟨from⟩}{⟨to⟩}{⟨i⟩}` does the same, starting from element at index `⟨i⟩` (the first element has index 0).

As an example again without questions:

Three digits from 2 to 4 and two letters: A 2 3 F 4.

Three digits and two letters: 2 8 4 E D.

Three digits and two letters: 4 E 2 5 A.

---

```

\element{digits}{ 1}\element{digits}{ 2}\element{digits}{ 3}
\element{digits}{ 4}\element{digits}{ 5}\element{digits}{ 6}
\element{digits}{ 7}\element{digits}{ 8}\element{digits}{ 9}
\element{letters}{ A}\element{letters}{ B}\element{letters}{ C}
\element{letters}{ D}\element{letters}{ E}\element{letters}{ F}

\shufflegroup{letters}
\cleargroup{mixed}
\copygroupfrom[3]{digits}{mixed}{1}\copygroup[2]{letters}{mixed}
\shufflegroup{mixed}
Three digits from 2 to 4 and two letters:\insertgroup{mixed}.

\shufflegroup{digits}\shufflegroup{letters}
\cleargroup{mixed}
\copygroup[3]{digits}{mixed}\copygroup[2]{letters}{mixed}
\shufflegroup{mixed}
Three digits and two letters:\insertgroup{mixed}.

\shufflegroup{digits}\shufflegroup{letters}
\cleargroup{mixed}
\copygroup[3]{digits}{mixed}\copygroup[2]{letters}{mixed}
\shufflegroup{mixed}
Three digits and two letters:\insertgroup{mixed}.

```

---

You can find an example involving questions in section [2](#).

### 3.5 Students identification

`\namefield` There are two ways to associate students to their sheets.

`\AMCcodeGrid`  
`\AMCcodeGridInt`

- Always add to one page of each copy some place for the student to write down his name. If you want AMC software to be able to cut the scan around this area to present it to you and ask you to read the written name (this is called manual association), you must use the `\namefield{<descr>}` command. The `<descr>` argument contains the  $\text{\LaTeX}$  code used to format the name field on the page. For example:

---

The name field

---

Name and surname:

.....

```

\namefield{\fbox{
  \begin{minipage}{15em}
    Name and surname:\vspace*{3ex}\par
    \noindent\dotfill\vspace{2mm}
  \end{minipage}
}}

```

---

You can see that the `\namefield` command has no effect on the produced document. In fact, its only purpose is to log in the `.xy` file information about the position of the name field on the page, to be used by the software analysing the scans.

- For automated student identification, if for example students have a 6-digits student number, you can ask them to code it somewhere on the question sheet. This can be done using the `\AMCcodeGridInt[⟨opts⟩]{⟨key⟩}{⟨ndigits⟩}` command, where `⟨key⟩` is the key identifier, that can be used to retrieve coded student numbers from the scans, and `⟨ndigits⟩` is the number of digits for numbers to be coded.

Student ID												
<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<input type="checkbox"/> 0	<code>\AMCcodeGridInt{student}{6}</code>						
<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1	<input type="checkbox"/> 1							
<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2	<input type="checkbox"/> 2							
<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3	<input type="checkbox"/> 3							
<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4	<input type="checkbox"/> 4							
<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5	<input type="checkbox"/> 5							
<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6	<input type="checkbox"/> 6							
<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7	<input type="checkbox"/> 7							
<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8	<input type="checkbox"/> 8							
<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9	<input type="checkbox"/> 9							

For smaller number of digits, the “horizontal” form can be preferred:

Student ID, horizontal form												
<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9			
<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9			
<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9			

`\AMCcodeGridInt[h]{student}{3}`

### 3.6 Separate answer sheet

`\AMCformBegin` To produce separate answer sheets as seen in section 2.2,

`\AMCform`

`\AMCcleardoublepage`

1. use the `separateanswersheet` package option.
2. use the `\AMCformBegin` command at the beginning of the answer sheet description. This command usually follows a command to get a new page. This command can be the classical `\clearpage` for single-sided question sheets, or the `\AMCcleardoublepage` command, that



go to the next odd numbered page, so that the answer sheet is on a separate sheet even when printing in duplex mode.

3. use the `\AMCform` command to insert all boxes for all questions.

See section 2.2 for an example.

### 3.7 Random computation questions

One can use the  $\text{\LaTeX}$  package `fp` to make random computation questions, as can be seen in the following example (don't forget to load package `fp`):

Random computation questions

**Question 6**      How much are 2 plus 8?

☐ 9      ☒ 10      ☐ 16      ☐ -6

```

\begin{question}{simplesum}
  \FPeval\VQa{trunc(1+random*8,0)}
  \FPeval\VQb{trunc(4+random*5,0)}
  \FPeval\VQsum{clip(VQa+VQb)}
  \FPeval\VQnoA{clip(VQa+VQb-1)}
  \FPeval\VQnoB{clip(VQa*VQb)}
  \FPeval\VQnoC{clip(VQa-VQb)}
  How much are \VQa{} plus \VQb{}?
  \begin{choiceshoriz}
    \correctchoice{\VQsum}
    \wrongchoice{\VQnoA}
    \wrongchoice{\VQnoB}
    \wrongchoice{\VQnoC}
  \end{choiceshoriz}
\end{question}

```

In this example, `\VQa` and `\VQb` are used to store two random integers (the first between 1 and 8, and the second between 4 and 8). Then `\VQsum` stores the sum of these two integers, and `\VQnoA`, `\VQnoB` and `\VQnoC` are other values that will be used as distractors in the multiple choice question.

**\AMCIntervals**      In some cases, command `\AMCIntervals{\langle x \rangle}{\langle x0 \rangle}{\langle x1 \rangle}{\langle \delta \rangle}` from `automultiplechoice` can be useful. It adds a sequence of choices made of intervals  $[x_i, x_i + \delta[$  of length  $\langle \delta \rangle$  covering the interval  $[\langle x0 \rangle, \langle x1 \rangle[$ , using `\correctchoice` when  $\langle x \rangle$  lies in the interval, and `\wrongchoice` otherwise.

Pick the right interval

**Question 7** Let  $X$  and  $Y$  be two independent random variables, following exponential laws with respective parameters 5 and 8. In which interval lies the probability  $P[X < Y]$ ?

- |                                     |  |                                     |                                     |                                     |
|-------------------------------------|--|-------------------------------------|-------------------------------------|-------------------------------------|
| <input type="checkbox"/> [0, 0.1[   | <input type="checkbox"/> [0.2, 0.3[            | <input type="checkbox"/> [0.4, 0.5[ | <input type="checkbox"/> [0.6, 0.7[ | <input type="checkbox"/> [0.8, 0.9[ |
| <input type="checkbox"/> [0.1, 0.2[ | <input checked="" type="checkbox"/> [0.3, 0.4[ | <input type="checkbox"/> [0.5, 0.6[ | <input type="checkbox"/> [0.7, 0.8[ | <input type="checkbox"/> [0.9, 1[   |

```
\begin{question}{inf-expo-indep}
  \FPeval\VQa{trunc(2 + random * 4,0)}
  \FPeval\VQb{trunc(6 + random * 5,0)}
  \FPeval\VQr{VQa/(VQa+VQb)}
  Let  $X$  and  $Y$  be two independent random variables, following
  exponential laws with respective parameters  $\VQa$  and  $\VQb$ .
  In which interval lies the probability  $\text{P}[X < Y]$ ?
  \begin{multicols}{5}
    \begin{choices}[o]
      \AMCIntervals{\VQr}{0}{1}{0.1}
    \end{choices}
  \end{multicols}
\end{question}
```

One can also use the `\AMCnumericChoices` command to ask the student to enter a numerical value as his answer, as in the following example:

Numeric choices

**Question 8**

Compute  $\sqrt{11}$  and round it with two digits after period.

	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input checked="" type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
	.									
<input checked="" type="checkbox"/> +	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input type="checkbox"/> 2	<input checked="" type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9
<input type="checkbox"/> -	<input type="checkbox"/> 0	<input type="checkbox"/> 1	<input checked="" type="checkbox"/> 2	<input type="checkbox"/> 3	<input type="checkbox"/> 4	<input type="checkbox"/> 5	<input type="checkbox"/> 6	<input type="checkbox"/> 7	<input type="checkbox"/> 8	<input type="checkbox"/> 9

```
\begin{questionmultx}{sqrt}
  \FPeval\VQa{trunc(5+random*15,0)}
  \FPeval\VQs{VQa^0.5}

  Compute  $\sqrt{\VQa}$  and round it with two digits after period.

  \AMCnumericChoices{\VQs}{digits=3,decimals=2,sign=true,
    borderwidth=0pt,backgroundcol=lightgray,approx=5}
\end{questionmultx}
```

Note the use of `questionmultx` environment: we need this question to be *multiple* as several boxes has to be ticked, but we can't say that *several answers are correct*, so we don't show the ♣.

Available options that can be used in the second argument of the `\AMCnumericChoices` command are the following ( $\langle bool \rangle$  can be `true` or `false`, and  $\langle color \rangle$  must be a color known by the `xcolor` package):

`digits= $\langle num \rangle$`  gives the number of digits to request (defaults to 3).

`decimals= $\langle num \rangle$`  gives the number of digits after period to request (defaults to 0). Note that when `decimals` is positive, the LaTeX package `fp` must be loaded.

`base= $\langle num \rangle$`  gives the base for digits and decimals (defaults to 10).

`significant= $\langle bool \rangle$`  if `true`, the numbers to code are the first *significant* digits from the first argument of `\AMCnumericChoices`. For example, the right answer to `\AMCnumericChoices{56945.23}{digits=2,significant=true}` is 57.

`exponent= $\langle num \rangle$`  gives the number of digits for the exponent, when requesting to enter the result in scientific notation.

`nozero= $\langle bool \rangle$`  if `true`, the choice 0 is removed for all digits. May be useful when `\AMCnumericChoices` is used to get a small ( $< 10$ ) positive value.

`sign= $\langle bool \rangle$`  requests (or not) a signed value (default to `true`).

`exposign= $\langle bool \rangle$`  requests (or not) a signed value of the exponent (default to `true`).

`strict= $\langle bool \rangle$`  if `true`, a box has to be ticked for every digit and for the sign. If `false`, if some digits has no ticked box, they will be set to zero. Defaults to `false`.

`vertical= $\langle bool \rangle$`  if `true`, each digit is represented on one raw. If `false` (default), each digit is represented on one line.

`expovertical= $\langle bool \rangle$`  if `true`, the mantissa is above the exponent. If `false` (default), the mantissa is beside the exponent.

`reverse= $\langle bool \rangle$`  if `true`, place higher values of the digits on the top in vertical mode (defaults to `true`).

`vhead= $\langle bool \rangle$`  if `true`, in vertical mode, a header is placed over all digits rows, made using the command `\AMCncontextVHead` that is originally defined as `\def\AMCncontextVHead#1{\emph{b#1}}`. This default value is useful to number the binary digits. Default value is `false`.

`Tvhead= $\langle text \rangle$`  A coma separated list as `{H,T,0,t,h,th}` for header in `vhead` vertical mode. Needs `vhead` to be set (defaults to the empty list `{}`).

`vheadunitindex= $\langle num \rangle$`  The index of the Ones place in the `Tvhead` list, counting from the right. If zero, changed to `decimals + 1` (defaults to zero). For example, if `Tvhead={H,T,0,t,h,th}`, should be set as `vheadunitindex=4` (at least if `decimals` is not set to 3).

`hspace= $\langle space \rangle$`  sets the horizontal space between boxes (defaults to `.5em`).

`vspace= $\langle space \rangle$`  sets the vertical space between boxes (defaults to `1ex`).

`borderwidth= $\langle space \rangle$`  sets the width of the frame around all the boxes (defaults to `1mm`).

`bordercol=<color>` sets the color of the frame (defaults to `lightgray`).

`backgroundcol=<color>` sets the background color (defaults to `white`).

`Tsign=<text>` sets the text to print at the top of the boxes to set the sign (Can also be redefined by `\def\AMCtextSign{<text>}`, and defaults to be empty).

`Tpoint=<text>` sets the text for the period. Can also be redefined by `\def\AMCdecimalPoint{<text>}`, and defaults to `\raisebox{1ex}{\bf .}`.

`Texponent=<text>` sets the text before the exponent. Can also be redefined by `\def\AMCexponent{<text>}`, and defaults to `\times 10^{\textasciicircum}`.

`scoring=<bool>` if `true`, a scoring strategy is given to AMC for this question. Defaults to `true`.

`scoreexact=<num>` gives the score for an exact answer (defaults to 2).

`exact=<num>` sets the maximal distance to the correct integer value (value without the decimal point) for an answer to be said *exact* and be rewarded to `scoreexact` points (defaults to 0).

`scoreapprox=<num>` gives the score for an approximative answer (defaults to 1).

`approx=<num>` sets the maximal distance to the correct integer value (value without the decimal point) for an answer to be said *approximative* and be rewarded to `scoreapprox` points (defaults to 0).

`scorewrong=<num>` gives the score for a wrong answer (defaults to 0).

`ignoreblank` can be used (only with number base 10) to ignore digits for which no box has been ticked. This way, ticking 5 for the first digit, no box for the second and 3 for the third digit will code the number 53, while this would have coded 503 without the `ignoreblank` option (because the default value for the second digit is 0).

`keepas=<name>` keeps the value entered by the student in variable `{<name>}`, for future use with `alsocorrect` in another question.

`alsocorrect=<expression>` gives another acceptable answer, that can be based on the values entered by the student in the previous questions.

The text added at the end of the questions using `\AMCnumericChoices` when not in the separate answer sheet (and when a separate answer sheet is requested by the `separateanswersheet` package option) can also be set redefining the `\AMCtextGoto` command, as:

```
\def\AMCtextGoto{\par{\bf\emph{Please code the answer on  
the separate answer sheet.}}}
```

### 3.8 French command names

For backward compatibility, some of `automultiplechoice` commands, environments and package option have their French counterpart. You can always use either the English command or the French equivalent. See table 1 for details.

## 3.9 Customisation

### 3.9.1 Boxes

`\AMCboxStyle` The command `\AMCboxStyle{<style>}` can be used to specify the shape, color and dimensions of the boxes to be ticked. The argument `<style>` is a coma-separated list of `<key>=<value>` pairs, with the following possible `<key>`s:

**shape** for the shape to be used: either **square** or **oval**. Note that if **oval** is used, the  $\text{\LaTeX}$  package **tikz** must be loaded.

**width** for the width of the boxes.

**height** for the height of the boxes.

**size** for the size of the boxes (sets **width** and **height**).

**down** for the length the boxes are to be moved down.

**rule** for the rule width.

**outsidesepp** for the distance between the box and the letter when printed outside the box.

**color** for the color (only the box that are to be filled by the students and will be used for data capture). Use something that will be understood by the **xcolor** package.

Default values are

```
\AMCboxStyle{shape=square,size=2.5ex,down=.4ex,
             rule=.5pt,outsidesepp=.1em,color=black}
```

Setting the box color allows to print the boxes with some color that won't disturb too much the data capture (for example red, but some light grey can also be considered).

Boxes styling	
<p><b>Question 9</b>   <math>2 + 2 =</math></p> <p>Ⓐ 1   Ⓑ 4   Ⓒ 10</p>	<pre>\AMCboxStyle{shape=oval,color=red} \begin{question}{sum}\$2+2={}\$ \begin{choiceshoriz}[o]   \wrongchoice{1}\correctchoice{4}   \wrongchoice{10} \end{choiceshoriz} \end{question}</pre>

### 3.9.2 Codes

One may adapt the codes rendering from `\AMCcodeGrid` to one's needs modifying the following lengths:

- `\AMCcodeHspace` is the amount of horizontal space between two columns of digits,
- `\AMCcodeVspace` is the amount of vertical space between two rows of digits,

Default values are `\AMCcodeHspace=.5em` `\AMCcodeVspace=.5em`

### 3.9.3 Answers

Environment `choicescustom` will make use of the three commands `\AMCbeginAnswer` (before the first answer), `\AMCendAnswer` (after the last answer) and `\AMCanswer{\langle box \rangle}{\langle text \rangle}` (for each answer) to format the answers. Redefining them properly, some different answers formatting can be achieved. However, this does not seem to work with non-trivial settings...

The `choicescustom` environment

---

**Question 10**       $2+2=$  ☐ 4      ☐ 3      ☐ 2      )

---

```

\begin{question}{add}
  \def\AMCbeginAnswer{\$ \Big{$}
  \def\AMCendAnswer{\$ \Big}{$}
  \def\AMCanswer#1#2{#1 #2\hfill}
  2+2=
  \begin{choicescustom}
    \correctchoice{4}
    \wrongchoice{2}
    \wrongchoice{3}
  \end{choicescustom}
\end{question}

```

## 4 Implementation

This package uses the following other packages:

```

1 \RequirePackage{xcolor} % \fcolorbox to fill (or not) a box
2 \RequirePackage{fancyhdr} % \pagestyle{empty}
3 \RequirePackage{bophook} % \AtBeginPage
4 \RequirePackage{xkeyval} % \setkeys
5 \RequirePackage{rotating} % \rotatebox
6 \RequirePackage{fancybox} % \boxput
7 \RequirePackage{expl3}
8 \RequirePackage{csvsimple}
9 \RequirePackage{environ}
10 % \end{macrocode}
11 %
12 % First, we read the options that can be given by AMC through the
13 % |jobname-config.tex| file:
14 % \begin{macrocode}
15 \InputIfFileExists{\jobname-config.tex}%
16 {\message{Loading configuration file...^^J}}{}

```

`\AMC@amclog` Informations about questions and choices will be logged to a file with extension `amc`, to be parsed later. Macro `\AMC@amclog` writes to this file.

```

17 \newwrite\AMC@logfile
18 \immediate\openout\AMC@logfile=\jobname.amc
19 \def\AMC@amclog#1{\immediate\write\AMC@logfile{#1}}
20 \def\AMCmessage#1{\AMC@amclog{\string\message{#1}}}

```

`\AMC@LR` Colours management can be faulty in right-to-left mode: in these situations, we will make use of `\LR` from package `bidi` to get back to left-to-right mode. `\AMC@LR` is `\LR` if `bidi` is loaded.

```
21 \AtBeginDocument{\@ifpackageloaded{bidi}{%  
22   \PackageInfo{automultiplechoice}{Package bidi loaded: using LR for boxes.}%  
23   \let\AMC@LR=\LR}%  
24 {\let\AMC@LR=\relax}}%
```

## 4.1 Variables

Counters and boolean variables defined here are internal and should not be modified by the user.

The package defines the following counters:

`\AMCload@counter` number of choices already loaded for current question.

`\AMCid@quest` current question ID number (see section 4.7).

`\AMCid@etud` current student sheet number.

`\AMCid@etudstart` starting student sheet number of the current `onecopy` bloc.

`\AMCid@check` current page checking number.

`\AMCid@etudfin` last student sheet number for the exam.

`\AMCnum@copies` number of exam sheets to produce.

It also defines the following switches:

`\ifAMC@ordre` if choices are never to be shuffled.

`\ifAMC@shuffleG` if groups shuffling is allowed.

`\ifAMC@fullGroups` if groups are always fully inserted by `\insertgroup` and fully copied by `\copygroup`, irrespective to the optional parameter.

`\ifAMC@correthead` if some correction header is to be printed at the beginning.

`\ifAMC@affichekeys` if questions keys are to be printed.

`\ifAMC@keyslines` if questions keys should be printed on a single line before the question text.

`\ifAMC@correc` if correct choices are to be checked on the produced document.

`\ifAMC@qbloc` if questions are to be included in  $\text{\LaTeX}$  boxes (so that they can't be splitted on two different pages).

`\ifAMC@asqbloc` if questions are to be included in  $\text{\LaTeX}$  boxes in the answer sheet (so that they can't be splitted on two different pages).

`\ifAMC@rbloc` if answers are to be included in  $\text{\LaTeX}$  boxes (so that they can't be splitted on two different columns for example).

`\ifAMC@textPos` if questions and answers positions are to be logged.

`\ifAMC@extractOnly` if the PDF is only built to extract questions and answers images.

`\ifAMCcomplete@multi` if a choice “None of these answers are correct.” is to be added to every multiple question.

`\ifAMCquestionNumber` if AMC should step up the question number for each new question.

`\ifAMC@calibration` if this L<sup>A</sup>T<sub>E</sub>X run is used to get page layouts.

`\ifAMC@plain` if `automultiplechoice` won’t try to load useful packages (`etex`, `environ`) that extend `automultiplechoice` capabilities.

`\ifAMCune@bonne` if there is at least one correct answer for the current question.

`\ifAMCtype@multi` if the current question is a multiple question.

`\ifAMC@watermark` if the document is a draft, not to be used for exam.

`\ifAMC@ensemble` if answers are to be given on a separate answers sheet.

`\ifAMC@inside@box` if a letter or digit is to be printed inside all boxes.

`\ifAMC@inside@digit` if digits are to be written inside boxes instead of letters (when using a separate answer sheet for example).

`\ifAMC@outside@box` if labels for boxes are to be printed outside the box on the answer sheet.

`\ifAMCformulaire@dedans` is true for questions inside separate answer sheet.

`\ifAMC@zoneformulaire` is true for codes (made by `\AMCcodeGrid`) inside separate answer sheet.

`\ifAMC@pagelayout` is true if the AMC page layout, with signs for scan analysis, is to be used.

`\ifAMC@postcorrect` corresponds to the use of the `postcorrect` package option.

`\ifAMC@automarks` corresponds to the use of the `automarks` package option.

`\ifAMC@invisible` is true is the DVI/PDF output is not important (used for example for scoring strategy extraction).

`\ifAMC@pdfform` is true if the output is a PDF form. This PDF will not be printed but will be filled by the students with a PDF reader and sent back to the teacher.

```

25 \newcount\AMCload@counter
26 \newcount\AMCid@quest\AMCid@quest=-1
27 \newcount\AMCid@check
28 \newcount\AMCid@etud\AMCid@etud=0
29 \newcount\AMCid@etudstart\AMCid@etudstart=0
30 \newcount\AMCid@etudfin
31 \newcount\AMCnum@copies

32 \newif\ifAMC@ordre\AMC@ordrefalse
33 \newif\ifAMC@shuffle\AMC@shuffleGtrue
34 \newif\ifAMC@fullGroups\AMC@fullGroupsfalse
35 \newif\ifAMC@correthead\AMC@corretheadfalse
36 \newif\ifAMC@affichekeys\AMC@affichekeysfalse
37 \newif\ifAMC@keyline\AMC@keylinefalse
38 \newif\ifAMC@correc\AMC@correcfalse

```



```

39 \newif\ifAMC@textPos\AMC@textPosfalse
40 \newif\ifAMC@extractOnly\AMC@extractOnlyfalse
41 \newif\ifAMC@qbloc\AMC@qblocfalse
42 \newif\ifAMC@asqbloc\AMC@asqblocfalse
43 \newif\ifAMC@rbloc\AMC@rblocfalse
44 \newif\ifAMC@complete@multi\AMC@complete@multifalse
45 \newif\ifAMC@questionNumber\AMC@questionNumbertrue
46 \newif\ifAMC@calibration\AMC@calibrationfalse
47 \newif\ifAMC@catalog\AMC@catalogfalse
48 \newif\ifAMC@plain\AMC@plainfalse
49 \newif\ifAMC@bonne
50 \newif\ifAMC@type@multi
51 \newif\ifAMC@watermark\AMC@watermarktrue
52 \newif\ifAMC@inside@box\AMC@inside@boxfalse
53 \newif\ifAMC@outside@box\AMC@outside@boxfalse
54 \newif\ifAMC@ensemble\AMC@ensemblefalse
55 \newif\ifAMC@inside@digit\AMC@inside@digitfalse
56 \newif\ifAMC@formulaire@dedans\AMC@formulaire@dedansfalse
57 \newif\ifAMC@zoneformulaire
58 \newif\ifAMC@pagelayout\AMC@pagelayouttrue
59 \newif\ifAMC@postcorrect\AMC@postcorrectfalse
60 \newif\ifAMC@automarks\AMC@automarksfalse
61 \newif\ifAMC@invisible\AMC@invisiblefalse
62 \newif\ifAMC@pdfform\AMC@pdfformfalse
63 \let\AMC@completeMulti=\AMC@complete@multitru
64 \let\AMC@noCompleteMulti=\AMC@complete@multifalse

```

`\AMCid@name` The package also defines command `\AMCid@name` to be the current question identifier key.

```
65 \def\AMCid@name{}
```

## 4.2 Dimensions

`\AMCformVSpace` The following dimensions can be modified by the user to adjust questions formatting:

`\AMCformHSpace` `\AMCformVSpace` is the amount of vertical space between two questions in a separate answer sheet.

`\AMCformHSpace` is the amount of horizontal space between two answers boxes in a separate answer sheet.

`\AMCinterIrep` is the amount of vertical space to be added between two answers.

`\AMCinterBrep` is the amount of vertical space between two boxed answers (see `\AMCBoxedAnswers` and `\ifAMC@rbloc`).

`\AMCinterIquest` is the amount of vertical space left after a question, in standard mode (without package option `box`).

`\AMCinterBquest` is the amount of vertical space left after a question, in 'boxed' mode (with package option `box`).

`\AMCpostNquest` is the amount of vertical space left after a numeric question.

`\AMCpostOquest` is the amount of vertical space left after an open question.

```

66 \newdimen\AMCformVSpace\AMCformVSpace=1.2ex
67 \newdimen\AMCformHSpace\AMCformHSpace=.3em
68 \newdimen\AMCinterIrep\AMCinterIrep=\z@
69 \newdimen\AMCinterBrep\AMCinterBrep=.5ex
70 \newdimen\AMCinterIquest\AMCinterIquest=\z@
71 \newdimen\AMCinterBquest\AMCinterBquest=3ex
72 \newdimen\AMCpostNquest\AMCpostNquest=1.5ex
73 \newdimen\AMCpostOquest\AMCpostOquest=7mm

```

### 4.3 Human readable sheet ID position

`\AMCidsPosition` The position of the human readable sheet ID, near the corresponding binary boxes, is set with the `\AMCidsPosition` command, in the form `\AMCidsPosition{pos=<position>,width=<width>,height=<height>}`, where *<position>* is one of *side* (default), *top* and *none*, *<width>* is the width of the box enclosing the ID (default value is 4cm), and *<height>* is the height of the box enclosing the ID (default value is 3ex).

```

74 \newif\ifAMCids@top
75 \newif\ifAMCids@side
76 \newdimen\AMCids@width
77 \newdimen\AMCids@height
78 \define@choicekey*{AMCids}{pos}[ \AMCidsVar\AMCidsVarN]{none,top,side}{%
79   \ifcase\AMCidsVarN\relax
80     \AMCids@topfalse\AMCids@sidefalse
81   \or
82     \AMCids@toptrue\AMCids@sidefalse
83   \or
84     \AMCids@topfalse\AMCids@sidetrue
85   \fi
86 }
87 \define@key{AMCids}{width}{\AMCids@width=#1}
88 \define@key{AMCids}{height}{\AMCids@height=#1}
89 \def\AMCidsPosition#1{\setkeys{AMCids}{#1}}
90 \AMCidsPosition{pos=side,width=4cm,height=3ex}

```

### 4.4 Localisation

In this section, some localised strings or commands are defined, for English, French and Spanish languages.

`\AMCtext` To modify these texts, you can use command `\AMCtext`. For example, `\AMCtext{draft}{<text>}` sets the text to be printed behind each page of a draft exam.

```

91 \def\AMCtext#1#2{\expandafter\def\cename{AMC@loc@#1}\endcsname{#2}}
92 \def\AMClocalized#1{\cename{AMC@loc@#1}\endcsname}

```

#### 4.4.1 English

Text indicating draft exams:

```

93 \def\AMC@loc@draft{DRAFT}

```

Message at page bottom when compiled out of AMC gui:

```

94 \def\AMC@loc@message{For your examination, preferably print
95   documents compiled from auto-multiple-choice.}

```

Announcing a question in a separate sheet (parameter #1 is the question number):

```
96 \def\AMC@loc@qf#1{\textbf{Question #1:}}
```

Announcing a question (parameter #1 is the question number and parameter #2 can be the multiple question symbol, or be empty):

```
97 \def\AMC@loc@q#1#2{\textbf{Question #1} #2}
```

Headers for corrected version and catalog:

```
98 \def\AMC@loc@corrected{Corrected}
```

```
99 \def\AMC@loc@catalog{Catalog}
```

Localization text for Explanation

```
100 \def\AMC@loc@explain{\textit{\textbf{Explanation: }}}}
```

Last choice added at the end for multiple questions when option `completemulti` is used:

```
101 \def\AMC@loc@none{None of these answers are correct.}
```

Word for 'question', singular and plural forms:

```
102 \def\AMC@loc@question{question}
```

```
103 \def\AMC@loc@questions{questions}
```

Default text to write in the students' name box:

```
104 \def\AMC@loc@namesurname{Name and surname:}
```

#### 4.4.2 Catalan

Catalan localisation is called with option `lang=CA`.

```
105 \def\AMC@loc@CA{
106   \def\AMC@loc@draft{PROJECTE}
107   \def\AMC@loc@message{Pel vostre examen, imprimiu preferiblement
108     els documents compilats amb l'ajuda de auto-multiple-choice.}
109   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
110   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
111   \def\AMC@loc@corrected{Correcci\`o}
112   \def\AMC@loc@catalog{Cat\`aleg}
113   \def\AMC@loc@explain{\textit{\textbf{Explicaci\`o : }}}}
114   \def\AMC@loc@none{Cap de les respostes \`es correcta.}
115   \def\AMC@loc@question{pregunta}
116   \def\AMC@loc@questions{preguntes}
117   \def\AMC@loc@namesurname{Nom i cognoms:}
118 }
```

#### 4.4.3 Dutch

Dutch localisation is called with option `lang=NL`.

```
119 \def\AMC@loc@NL{
120   \def\AMC@loc@draft{Ontwerp}
121   \def\AMC@loc@message{Gebruik bij uw proefwerk bij voorkeur die
122     documenten welke door auto-multiple-choice zijn aangemaakt.}
123   \def\AMC@loc@qf##1{\textbf{Vraag ##1 :}}
124   \def\AMC@loc@q##1##2{\textbf{Vraag ##1} ##2}
125   \def\AMC@loc@corrected{Correctie}
126   \def\AMC@loc@catalog{Catalogus}
127   \def\AMC@loc@none{Geen van de antwoorden is juist.}
128   \def\AMC@loc@question{vraag}
```

```

129 \def\AMC@loc@questions{vragen}
130 \def\AMC@loc@namesurname{Achternaam en voornaam:}
131 }

```

#### 4.4.4 French

French localisation is called with option `francais`, or `lang=FR`.

```

132 \def\AMC@loc@FR{
133   \def\AMC@loc@draft{PROJET}
134   \def\AMC@loc@message{Pour votre examen, imprimez de pr\`ef\`erence
135     les documents compil\`es \`a l'aide de auto-multiple-choice.}
136   \def\AMC@loc@qf##1{\textbf{Question ##1 :}}
137   \def\AMC@loc@q##1##2{\textbf{Question ##1} ##2}
138   \def\AMC@loc@corrected{Correction}
139   \def\AMC@loc@catalog{Catalogue}
140   \def\AMC@loc@explain{\textit{\textbf{Explication : }}}
141   \def\AMC@loc@none{Aucune de ces r\`eponses n'est correcte.}
142   \def\AMC@loc@question{question}
143   \def\AMC@loc@questions{questions}
144   \def\AMC@loc@namesurname{Nom et pr\`enom :}
145 }

```

#### 4.4.5 German

German localisation is called with option `lang=DE`.

```

146 \def\AMC@loc@DE{
147   \def\AMC@loc@draft{ENTWURF}
148   \def\AMC@loc@message{Benutzen Sie f\"ur Ihre Pr\"ufung bevorzugt Dokumente die mit
149     auto-multiple-choice erstellt wurden.}
150   \def\AMC@loc@qf##1{\textbf{Frage ##1 :}}
151   \def\AMC@loc@q##1##2{\textbf{Frage ##1} ##2}
152   \def\AMC@loc@corrected{Korrektur}
153   \def\AMC@loc@catalog{Katalog}
154   \def\AMC@loc@explain{\textit{\textbf{Erkl\"arung : }}}
155   \def\AMC@loc@none{Keine dieser Antworten ist korrekt.}
156   \def\AMC@loc@question{Frage}
157   \def\AMC@loc@questions{Fragen}
158   \def\AMC@loc@namesurname{Vor- und Nachname:}
159 }

```

#### 4.4.6 Italian

Italian localisation is called with option `lang=IT`.

```

160 \def\AMC@loc@IT{
161   \def\AMC@loc@draft{BOZZA}
162   \def\AMC@loc@message{Per l'esame, \`e preferibile stampare i documenti
163     a partire da auto-multiple-choice.}
164   \def\AMC@loc@qf##1{\textbf{Domanda ##1:}}
165   \def\AMC@loc@q##1##2{\textbf{Domanda ##1} ##2}
166   \def\AMC@loc@corrected{Correzione}
167   \def\AMC@loc@catalog{Catalogo}
168   \def\AMC@loc@none{Nessuna risposta \`e giusta.}
169   \def\AMC@loc@question{domanda}

```

```

170 \def\AMC@loc@questions{domande}
171 \def\AMC@loc@namesurname{Nome e cognome:}
172 }

```

#### 4.4.7 Norwegian

Norwegian localisation is called with option `lang=NO`.

```

173 \def\AMC@loc@NO{
174   \def\AMC@loc@draft{UTKAST}
175   \def\AMC@loc@message{Det anbefales {\aa} skrive ut dokumentet
176   for gjennomgang \direkte fra auto-multiple-choice.}
177   \def\AMC@loc@qf##1{\textbf{Oppgave ##1 :}}
178   \def\AMC@loc@q##1##2{\textbf{Oppgave ##1} ##2}
179   \def\AMC@loc@corrected{Rettet}
180   \def\AMC@loc@catalog{Katalog}
181   \def\AMC@loc@none{Ingen svar er riktige.}
182   \def\AMC@loc@question{oppgave}
183   \def\AMC@loc@questions{oppgave}
184   \def\AMC@loc@namesurname{Etternavn og fornavn:}
185 }

```

#### 4.4.8 Portuguese

Portuguese localisation is called with option `lang=PT`.

```

186 \def\AMC@loc@PT{
187   \def\AMC@loc@draft{RASCUNHO}
188   \def\AMC@loc@message{Para o seu exame, use preferencialmente documentos compilados do auto-multiple-choice.}
189   \def\AMC@loc@qf##1{\textbf{Quest\~ao ##1:}}
190   \def\AMC@loc@q##1##2{\textbf{Quest\~ao ##1} ##2}
191   \def\AMC@loc@corrected{Corrigido}
192   \def\AMC@loc@catalog{Cat\'alogo}
193   \def\AMC@loc@explain{\textit{\textbf{Justifique: }}}}
194   \def\AMC@loc@none{Nenhuma das respostas apresentadas est\'a correta.}
195   \def\AMC@loc@question{Quest\~ao}
196   \def\AMC@loc@questions{Quest\~oes}
197   \def\AMC@loc@namesurname{Nome e apelido:}
198 }

```

#### 4.4.9 Spanish

Spanish localisation is called with option `lang=ES`.

```

199 \def\AMC@loc@ES{
200   \def\AMC@loc@draft{BORRADOR}
201   \def\AMC@loc@message{Para revisi\'on, preferentemente imprimir documentos compilados
202   desde auto-multiple-choice.}
203   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
204   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
205   \def\AMC@loc@corrected{Correcci\'on}
206   \def\AMC@loc@catalog{Cat\'alogo}
207   \def\AMC@loc@none{Ninguna de estas preguntas son correctas.}
208   \def\AMC@loc@question{pregunta}
209   \def\AMC@loc@questions{preguntas}
210   \def\AMC@loc@namesurname{Nombre y apellidos:}

```



`\AMC@Randomseed` The seed of this generator is set to 1515, but another value can be given using the command `\AMC@Randomseed{<seed>}`.

```
243 \AMC@SRset{1515}
244 \def\AMC@Randomseed#1{\AMC@SRset{#1}}
```

#### 4.6.2 Uniform random deviates

`\AMC@SRnextByte` This generator is used to build first a 20-bit uniform integer generator (macro `\AMC@SRnextByte`).  
`\AMC@SRmax` Then, using modulo, a (nearly) uniform generator on  $\{0, \dots, n-1\}$  is built: command `\AMC@SRmax{n}` puts in `\AMC@SR@count` the random deviate.

```
245 \newcount\AMC@SR@count
246 \def\AMC@SR@time{\AMC@SRset{\time}}
247 \newcount\AMC@SRnum
248 \def\AMC@SRnextByte{\AMC@SRnum=\z@%
249   \AMC@SR@count=20%
250   \loop\multiply\AMC@SRnum\tw@%
251     \AMC@SRtest{\advance\AMC@SRnum\@ne}{}%
252   \ifnum\AMC@SR@count>\@ne\advance\AMC@SR@count\m@ne\repeat%
253 }
254 \newcommand\AMC@SRmax[1]{\AMC@SRnextByte%
255   \AMC@SR@count=\AMC@SRnum%
256   \divide\AMC@SR@count by #1\relax%
257   \multiply\AMC@SR@count by #1\relax%
258   \advance\AMC@SRnum by -\AMC@SR@count%
259 }
```

#### 4.6.3 Tokens shuffling

`\AMCsw@p` The package defines the macro `\AMCsw@p` to swap the values of two token registers given as  
`\AMC@shuffletoks` parameters.

After defining  $n$  token registers `\foo@i`, `\foo@ii`, `\foo@iii`, `\foo@iv` and so on, you can shuffle them using `\AMC@shuffletoks[<a>]{<n>}{<foo>}`. With optional argument `<a>`, registers are shuffled from number `<a>` to `<n>` (default value for `<a>` is 1).

```
260 \newcount\AMC@sti
261 \newcount\AMC@stil
262 \newtoks\AMCsw@p@
263 \newcommand\AMCsw@p[2]{%
264   \global\AMCsw@p@=#1%
265   \global#1=#2%
266   \global#2=\AMCsw@p@}
267 \newcommand{\AMC@shuffletoks}[3][\@ne]{%
268   \AMC@sti=#2\relax%
269   \AMC@stil=#2\relax%
270   \advance\AMC@stil\@ne%
271   \advance\AMC@stil -#1\relax%
272   \@whilenum\AMC@sti>#1\do{%
273     \AMC@SRmax{\AMC@stil}\advance\AMC@SRnum #1\relax%
274     \AMCsw@p{\csname #3\romannumeral\AMC@SRnum\endcsname}%
275       {\csname #3\romannumeral\AMC@sti\endcsname}%
276     \advance\AMC@sti\m@ne\relax%
277     \advance\AMC@stil\m@ne\relax%
278   }}
```

## 4.7 Keys numbering

`\AMC@unnumero` This package allocates a unique integer ID to each question key from the questionnaire.  
`\AMC@affecte` The counter `\AMC@numerotation` keeps track of the number of keys which already had an ID. Command `\AMC@definitnumero{n}{key}` allocates ID  $n$  to the key `key`. Command `\AMC@prepare{key}` looks if an ID had already been associated to `key`, and, if not, makes a new ID allocation for `key`. Command `\AMC@unnumero{key}` returns the ID associated with `key` (creating one if necessary). Command `\AMC@affecte{key}{\cnt}` give to counter `\cnt` the value of the ID associated to `key` (creating one if necessary).

```
279 \newcount\AMC@numerotation\AMC@numerotation=\z@%
280 \def\AMC@definitnumero#1#2{\AMCmessage{NUM=#1=#2}%
281   \expandafter\global\expandafter\def\csname AMC@numtab@#2\endcsname{#1}}
282 \def\AMC@prepare#1{\expandafter\ifx\csname AMC@numtab@#1\endcsname\relax%
283   \global\advance\AMC@numerotation\@ne%
284   \expandafter\AMC@definitnumero\expandafter{\the\AMC@numerotation}{#1}\fi}
285 \def\AMC@unnumero#1{\AMC@prepare{#1}\csname AMC@numtab@#1\endcsname}
286 \def\AMC@affecte#1#2{\AMC@prepare{#1}\global#2=\csname AMC@numtab@#1\endcsname}
```

## 4.8 Boxes

### 4.8.1 Character logging

`\AMC@logchar` The command `\AMC@logchar{<char>}{<key>}` logs the character written in the box referenced as `<key>` in the .amc file. This is used in catalog mode, to get understandable references to answers from the statistics tables of the ODS export.

```
287 \def\AMC@logchar#1#2{%
288   \protected@write\AMC@logfile{}{%
289     \string\answer%
290     {\the\AMCid@etud/\thepage:#2}%
291     {#1}}%
292 }
```

### 4.8.2 Position logging

`\AMC@tracebox` Command `\AMC@tracebox{<trace>}{<key>}{<content>}` makes a L<sup>A</sup>T<sub>E</sub>X box around `<content>`,  
`\AMC@pagepos` and, if `<trace>` is not empty, logs to the .xy file informations to be able to compute exact location of this box on the page, attached to the box identification `<key>`.

Command `\AMC@pagepos` logs page and page size informations at the beginning of each page.

```
293 \def\AMC@shapename@{\ifAMC@invisible none\else\AMC@shapename\fi}
294 \def\AMC@tracepos#1#2{%
295   \ifAMC@calibration\ifx\@empty#1\@empty\else%
296     \pdfsavepos\protected@write\AMC@XYFILE{}{%
297       \string\tracepos%
298       {\the\AMCid@etud/\thepage:#2}%
299       {\noexpand\number\pdflastxpos sp}%
300       {\noexpand\number\pdflastypos sp}%
301       {\AMC@shapename}}%
302   \fi\fi}
303 \def\AMC@traceposx#1#2{%
304   \ifAMC@calibration\ifx\@empty#1\@empty\else%
305     \pdfsavepos\protected@write\AMC@XYFILE{}{%
306       \string\tracepos%
```



```

307     {\the\AMCid@etud/\thepage:#2}%
308     {\noexpand\number\pdflastxpos sp}%
309     {0sp}%
310     {\AMC@shapename}}%
311   \fi\fi}
312 \def\AMC@traceposy#1#2{%
313   \ifAMC@calibration\ifx\@empty#1\@empty\else%
314     \pdfsavepos\protected@write\AMC@XYFILE{}\{%
315       \string\tracepos%
316       {\the\AMCid@etud/\thepage:#2}%
317       {0sp}%
318       {\noexpand\number\pdflastypos sp}%
319       {\AMC@shapename}}%
320   \fi\fi}
321 \newcommand\AMC@tracebox[3]{%
322   \vbox{\AMC@traceposy{#1}{#2}%
323     \hbox{\AMC@traceposx{#1}{#2}#3\AMC@traceposx{#1}{#2}}%
324     \AMC@traceposy{#1}{#2}}}%
325 \def\AMC@pagepos{%
326   \ifAMC@calibration\protected@write\AMC@XYFILE{}\{%
327     \string\page%
328     {\the\AMCid@etud/\thepage/\the\AMCid@check}%
329     {\the\paperwidth}{\the\paperheight}%
330     {\the\pdfpagewidth}{\the\pdfpageheight}}\fi}

```

`\AMCdontScan` The commands `\AMCdontScan`, `\AMCdontAnnotate` and `\AMCretick` write into the xy file instructions related to the current question.

`\AMCretick` 331 \newcommand{\AMCdontScan}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontscan{\the\AMCid@etud/  
332 \newcommand{\AMCdontAnnotate}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontannotate{\the\AMCid@etud/  
333 \newcommand{\AMCretick}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\retick{\the\AMCid@etud,\thepage:  
334 %

`\AMC@tracechar` The macro `\AMC@tracechar{<char>}{<unused>}{<trace>}{<key>}` is used to log (for further processing with AMC), into to .xy file, the character used to identify the box.

```

335 \newcommand\AMC@tracechar[4]{%
336   \ifAMC@calibration\ifx\@empty#3\@empty\else%
337     \protected@write\AMC@XYFILE{}\{%
338       \string\boxchar{\the\AMCid@etud/\thepage:#4}{#1}%
339     }%
340   \fi\fi%
341 }

```

`amcxyfile (env.)` The following lines defines an environment to tag positions outputs for a particular part of the document. This is used mainly for documentation or testing.

```

342 \newenvironment{amcxyfile}[1]{%
343   \protected@write\AMC@XYFILE{}\{\string\xyopen{#1}}%
344 }{%
345   \protected@write\AMC@XYFILE{}\{\string\xyclose{}}%
346 }

```

`\AMCzone` The `\AMCzone[<flags>]{<zone name>}{<zone content>}` is a simple call to `\AMC@tracebox`:

```

347 \newcommand\AMCzone[3][\AMC@tracebox]{1}{__zone:#1:#2}{#3}}

```

`\namefield` The `\namefield{<name field content>}` is a simple call to `\AMCzone`:

```
348 \newcommand{\namefield}[2][id]{\AMCzone[#1]{__n}{#2}}
```

It is used to enclose the page region where students are to write their names, so as to retrieve it easily from the scans.

`\namefielddots` The command `\namefielddots` can be used to fill a line with dots (printed sheets) or use a text field in PDF forms:

```
349 \newcommand{\namefielddots}{%
350   \noindent%
351   \ifAMC@pdfform%
352     \hspace*{\fill}%
353     \TextField[name={\the\AMCid@etud:namefield},width=.95\linewidth,bordercolor=0 0 0]{}%
354     \hspace*{\fill}
355   \else%
356     \dotfill
357   \fi%
358 }
```

As an example,

```
\namefield{\fbox{%
  \begin{minipage}{5cm}
    Name:

    \vspace*{.5cm}
    \namefielddots
    \vspace{2mm}
  \end{minipage}}}
```

produces the following box:

Name:  
  
 .....

and outputs information about the position of the box in the `.xy` file, as seen in section 5.1.

### 4.8.3 Boxes to be checked by students

`\AMC@answerBox@` There are two styles for boxes to be checked by the students. The first one is an empty box, printed beside the answer. The second is a box with a character in it. It is mainly used when answers are to be given on a separate answer sheet.

These boxes can be drawn using command `\AMC@answerBox@{<char>}{<answer>}{<trace>}{<key>}`: `<char>` is the character to print inside the box, `<trace>` is non-empty if you want to log the box position in the `.xy` file, `<key>` is the box identification, and `<answer>` is an answer to be written in the box (or `\AMC@checkbox` for filling the box).

Depending on the required shape for the boxes, the corresponding

`\AMC@shape@xxx{<char>}{<answer>}{<trace>}{<key>}`

command is used.

- `\AMC@answerBox@{K}{1}{test}` produce the box  $\boxed{K}$ , writing the lines in the .xy file shown in section 5.2.
- `\AMC@answerBox@{K}{\AMC@checkedbox}{}` produces  $\boxed{\blacksquare}$
- `\AMC@answerBox@{}{8}{}` produces  $\boxed{8}$
- `\AMC@answerBox@{K}{8}{1}{testb}` produces  $\textcircled{8}$  with `\AMCboxStyle{shape=oval,color=red}`

```

359 \def\AMC@checkedbox{}
360 \let\AMC@new@savebox=\newsavebox
361 \let\AMC@save@box=\savebox
362 \let\AMC@use@box=\usebox
363 \newif\ifAMC@draw@cross

```

The `\AMC@smashcentered{<text>}` command shows the `<text>` centered at point.

```

364 \newbox\AMC@smashbox
365 \newdimen\AMC@smashboxheight
366 \newcommand{\AMC@smashcentered}[1]{%
367   \setbox\AMC@smashbox\hbox{#1}%
368   \AMC@smashboxheight=\ht\AMC@smashbox%
369   \advance\AMC@smashboxheight by \dp\AMC@smashbox%
370   \vfuzz=\AMC@smashboxheight\hfuzz=\wd\AMC@smashbox%
371   \hspace*{-.5\wd\AMC@smashbox}\hbox to .5\wd\AMC@smashbox{%
372     \vbox to 0pt{%
373       \vspace*{-.5\AMC@smashboxheight}\vbox to .5\AMC@smashboxheight{%
374         \box\AMC@smashbox}}}%
375 }%

```

`\AMC@setcolors@{<trace>}{<answer>}` sets colours `\AMC@boxcolor@` and `\AMC@fillcolor@` according to its arguments. It also sets the `\ifAMC@draw@cross` switch if AMC should draw a cross instead of filling the box.

```

376 \newcommand\AMC@setcolors@[2]{%
377   \def\AMC@boxcolor@{\AMC@boxcolor}%
378   \ifx\@empty#1\@empty \def\AMC@boxcolor@{black}\fi%
379   \ifAMC@correc\def\AMC@boxcolor@{black}\fi%
380   \def\AMC@fillcolor@{\ifx #2\AMC@checkedbox%
381     \AMC@boxcolor@\else white\fi}%
382   \AMC@draw@crossfalse%
383   \ifKV@AMCdim@cross\ifx #2\AMC@checkedbox%
384     \AMC@draw@crosstrue\fi\fi%
385 }
386 \newcommand\AMC@answerBox@[4]{%
387   \ifAMC@catalog%
388     \AMC@logchar{#1}{#4}%
389     \fi%
390     \AMC@LR{\hspace{0pt}%
391       \lower\AMC@boxeddown\hbox{\csname AMC@shape@\AMC@shapename@\endcsname%
392         {\AMCchoiceLabelFormat{#1}}{#2}{#3}{#4}}}%
393 }
394 \newcommand\AMC@shapeprepare@square{}
395 \newcommand\AMC@shape@square[4]{%
396   \fboxsep=\z@\fboxrule=\AMC@boxedrule%
397   \AMC@setcolors@{#3}{#2}%
398   \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%

```

```

399 \fcolorbox{\AMC@boxcolor@}{\AMC@fillcolor@}%
400 {%
401   \boxput*(0,0){%
402     \ifAMC@draw@cross\AMC@crosschar\fi%
403   }{%
404     \vbox to \AMC@boxedheight{%
405       \AMC@tracepos{#3}{#4}%
406       \vfill%
407       \hbox to \AMC@boxedwidth{\hfill%
408         \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
409         \AMC@smashcentered{#2}%
410         \hfill}\vfill}}%
411     \AMC@tracepos{#3}{#4}}%
412 }

```

`\AMC@makeovalbox{<trace>}{<answer>}{<box>}` prepares an oval frame in the L<sup>A</sup>T<sub>E</sub>X box `<box>`.

```

413 \newcommand\AMC@makeovalbox[3]{%
414   \AMC@setcolors@{#1}{#2}%
415   \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
416   \AMC@save@box{#3}{%
417     \begin{tikzpicture}%
418       \useasboundingbox (-0.5\AMC@boxedwidth-0.5\AMC@boxedrule,0.5\AMC@boxedheight+0.5\AMC@boxedrule)
419       rectangle (0.5\AMC@boxedwidth+0.5\AMC@boxedrule,-0.5\AMC@boxedheight-0.5\AMC@boxedrule);
420       \draw[\AMC@boxcolor@,fill=\AMC@fillcolor@,line width=\AMC@boxedrule,rounded corners=\AMC@oval@radi
421       (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight)
422       rectangle (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
423       \ifAMC@draw@cross
424         \draw[\AMC@boxcolor@,line width=\AMC@crossrule]
425         (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight)
426         (0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (-0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
427       \fi
428     \end{tikzpicture}}%
429 }
430 \newcommand\AMC@shapeprepare@oval[%
431   \ifx\AMC@ovalbox@R\@undefined\else%
432     \AMC@makeovalbox{1}{\AMC@ovalbox@R}%
433     \AMC@makeovalbox{1}{\AMC@checkedbox}{\AMC@ovalbox@RF}%
434     \AMC@makeovalbox{}{\AMC@ovalbox@}%
435     \AMC@makeovalbox{}{\AMC@checkedbox}{\AMC@ovalbox@F}%
436   \fi%
437 }
438 \newcommand\AMC@shape@oval[4]{%
439   \AMC@setcolors@{#3}{#2}%
440   \AMC@tracebox{#3}{#4}{\boxput*(0,0){%
441     \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
442     \AMC@smashcentered{#2}%
443   }{%
444     \ifx\@empty#3\@empty%
445       \ifx #2\AMC@checkedbox%
446         \AMC@use@box{\AMC@ovalbox@F}%
447       \else%
448         \AMC@use@box{\AMC@ovalbox@}%
449       \fi%
450     \else%

```

```

451     \ifx #2\AMC@checkbox%
452       \AMC@use@box{\AMC@ovalbox@RF}%
453     \else%
454       \AMC@use@box{\AMC@ovalbox@R}%
455     \fi%
456   \fi%
457 }}%
458 }
459 \newcommand\AMC@shapeprepare@form{}
460 \newcommand\AMC@shape@form@base[5]{%
461   \ifx #2\AMC@checkbox%
462     \def\AMC@shape@form@ticked{true}%
463   \else%
464     \def\AMC@shape@form@ticked{false}%
465   \fi%
466   \AMC@tracebox{#3}{#4}{%
467     \CheckBox[checked=\AMC@shape@form@ticked,%
468       checkboxsymbol=\ding{110},name={#5},%
469       bordercolor=0 0 0,%
470       width=\AMC@boxedwidth,height=\AMC@boxedheight]{#5}%
471   }%
472 }
473 \newcommand\AMC@shape@form[4]{%
474   \AMC@shape@form@base{#1}{#2}{#3}{#4}{\the\AMCid@etud:#4}%
475 }
476 \newcommand\AMC@shapeprepare@none{}
477 \newcommand\AMC@shape@none[4]{ #1 }

```

**\AMC@answerBox** Command \AMC@answerBox is the same as \AMC@answerBox@, but if  $\langle char \rangle$  is empty, it is replaced by an arabic or alphabetical counter, depending on the use of the **digits** package option.

**\AMCchoiceLabel** To use another way to label the choices boxes, the user can redefine the \AMCchoiceLabel macro, which takes as argument the name of the counter used to number the choices. One can for example use \def\AMCchoiceLabel#1{\alph{#1}} to ask for lowercase letters.

To write these labels with another font, size, or so, the user can redefine the \AMCchoiceLabelFormat macro, which takes as argument the label. One can for example get sans serif bold labels with \def\AMCchoiceLabelFormat#1{\textsf{\textbf{#1}}}.

```

478 \def\AMCchoiceLabel#1{%
479   \ifAMC@inside@digit\arabic{#1}%
480   \else\Alph{#1}\fi%
481 }
482 \def\AMCchoiceLabelFormat#1{#1}
483 \newcounter{AMC@ncase}
484 \setcounter{AMC@ncase}{0}
485 \newcommand\AMC@answerBox[4]{%
486   \AMC@answerBox@{\ifx\@empty#1\@empty%
487     \AMCchoiceLabel{AMC@ncase}%
488     \else #1\fi}{#2}{#3}{#4}}

```

**\AMCboxStyle** The dimensions of these box are managed by \AMCboxDimensions{ $\langle sizes \rangle$ }, where  $\langle sizes \rangle$  is a coma separated list of  $\langle name \rangle = \langle dimension \rangle$  constructs. Here,  $\langle name \rangle$  can be **size** for the box size, **rule** for the box rule width, **down** for moving the box down, **color** for the box color and **outsidesep** for the distance between the box and the letter (when outside the box).

The  $\langle color \rangle$  value given to `color` is a color that should be defined for the `xcolor` package. This color is used only in the case the box will be used for data capture: it is not used on the corrected answer sheet (`answers` or `indivanswers` package option), and not used on the subject part of an exam with a separate answer sheet (`separateanswersheet` package option).

The `\AMCboxColor{ $\langle color \rangle$ }` command is defined as an alias to `\AMCboxStyle{color= $\langle color \rangle$ }`, and `\AMCboxDimensions` as an alias to `\AMCboxStyle`, for backward compatibility.

```

489 \newlength\AMC@boxedrule
490 \newlength\AMC@crossrule
491 \newlength\AMC@boxeddown
492 \newlength\AMC@boxedwidth
493 \newlength\AMC@boxedheight
494 \newlength\AMC@oval@radius
495 \newlength\AMC@outside@sep
496 \define@choicekey{AMCdim}{shape}{square,oval,form,none}{\def\AMC@shapename{#1}}
497 \define@key{AMCdim}{size}{\AMC@boxedwidth=#1\AMC@boxedheight=#1}
498 \define@key{AMCdim}{height}{\AMC@boxedheight=#1}
499 \define@key{AMCdim}{width}{\AMC@boxedwidth=#1}
500 \define@key{AMCdim}{rule}{\AMC@boxedrule=#1}
501 \define@key{AMCdim}{outsidesep}{\AMC@outside@sep=#1}
502 \define@key{AMCdim}{down}{\AMC@boxeddown=#1}
503 \define@key{AMCdim}{color}{\def\AMC@boxcolor{#1}}
504 \define@boolkey{AMCdim}{cross}[false]{ }
505 \define@key{AMCdim}{crosschar}{\textbf{\textsf{X}}}{\def\AMC@crosschar{#1}}
506 \define@key{AMCdim}{crossrule}[1.5pt]{\AMC@crossrule=#1}
507 \def\AMC@shapeprepare{\csname AMC@shapeprepare@\AMC@shapename@ \endcsname}
508 \def\AMCboxStyle#1{%
509   \setkeys{AMCdim}{#1}%
510   \ifnum\AMC@boxedwidth<\AMC@boxedheight%
511     \AMC@oval@radius=\AMC@boxedwidth\divide\AMC@oval@radius\tw@%
512   \else%
513     \AMC@oval@radius=\AMC@boxedheight\divide\AMC@oval@radius\tw@%
514   \fi%
515   \AMC@shapeprepare%
516 }
517 \AMCboxStyle{shape=square,size=2.5ex,down=.4ex,rule=.5pt,outsidesep=.1em,color=black,cross,crosschar,crossrule}
518 \newcommand\AMCboxColor[1]{\AMCboxStyle{color=#1}}
519 \let\AMCboxDimensions=\AMCboxStyle

```

`\AMCboxOutsideLetter` Command `\AMC@box{ $\langle char \rangle$ }{ $\langle answer \rangle$ }` prints a box with character  $\langle char \rangle$  inside, showing answer  $\langle answer \rangle$  (`\AMC@checkbox` to get a filled box), using global variables to identify the box (`\AMC@formBox@` (question and choice).

`\AMC@formBox` It calls `\AMC@formBox@{ $\langle char \rangle$ }{ $\langle answer \rangle$ }{ $\langle trace \rangle$ }{ $\langle key \rangle$ }` to actually render the box.

`\AMC@formBox@` Command `\AMC@formBox` simply sets the first argument when empty before calling `\AMC@formBox@`.

The command `\AMCboxOutsideLetter{ $\langle box \rangle$ }{ $\langle char \rangle$ }` is called to print the box *and* the character  $\langle char \rangle$  outside (and next to) it. The character is formatted using `\AMCoutsideLabelFormat` first: if you need bold characters, redefine it with `\def\AMCoutsideLabelFormat#1{\textbf{#1}}`

`\AMC@keyBox@` is used instead of `\AMC@formBox@` when the text that corresponds to the answer is the letter/character inside the box itself (see `\AMCcodeGrid` and `\AMCnumericChoices`).

```

520 \def\AMCoutsideLabelFormat#1{#1}
521 \newcommand\AMCboxOutsideLetter[2]{#1\nobreak\hspace{.1em}\AMCoutsideLabelFormat{#2}}
522 \newif\ifAMC@printformoutside@
523 \newcommand\ifAMC@printformoutside{%

```

```

524 \AMC@printformoutside@false%
525 \ifAMC@ensemble\ifAMC@outside@box%
526 \ifAMCformulaire@dedans\AMC@printformoutside@true\fi%
527 \ifAMC@zoneformulaire\AMC@printformoutside@true\fi%
528 \fi\fi%
529 \ifAMC@printformoutside@%
530 }
531 \newcommand\AMC@formBox@[4]{%
532 \ifAMC@printformoutside% letter to be written outside the box
533 \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
534 \else%
535 \AMC@answerBox@{#1}{#2}{#3}{#4}%
536 \fi%
537 \AMC@tracechar{#1}{#2}{#3}{#4}%
538 }
539 \newif\ifAMC@printkeyoutside@%
540 \newcommand\ifAMC@printkeyoutside{%
541 \AMC@printkeyoutside@false%
542 \ifAMC@ensemble%
543 \ifAMC@outside@box\AMC@printkeyoutside@true\fi%
544 \else%
545 \ifAMC@inside@box\else\AMC@printkeyoutside@true\fi%
546 \fi%
547 \ifAMC@printkeyoutside@%
548 }
549 \newcommand\AMC@keyBox@[4]{%
550 \ifAMC@printkeyoutside%
551 \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
552 \else%
553 \AMC@answerBox@{#1}{#2}{#3}{#4}%
554 \fi%
555 \AMC@tracechar{#1}{#2}{#3}{#4}%
556 }
557 \newcommand\AMC@formBox[4]{%
558 \AMC@formBox@{\ifx\@empty#1\@empty%
559 \AMCchoiceLabel{AMC@ncase}%
560 \else #1\fi}{#2}{#3}{#4}%
561 }
562 \newcommand{\AMC@box}[2]{%
563 \ifAMC@ensemble%
564 \ifAMC@zoneformulaire% for codes inside form sheet
565 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
566 \else%
567 \ifAMCformulaire@dedans% for answer boxes inside form sheet
568 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
569 \else% outside form sheet: not to be read during data capture
570 \AMC@formBox{#1}{#2}{1}{casequestion:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
571 \fi\fi%
572 \else% no separate sheet for answers: always read
573 \ifAMC@inside@box%
574 \AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
575 \else%
576 \AMC@formBox@{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%

```

```

577 \fi%
578 \fi%
579 }

```

#### 4.8.4 Scoring zones

**\AMCscoreZone** The source file can define zones that will be used to print scores when annotating the completed answer sheets. The command `\AMCscoreZone{<zone>}` logs these zones positions on the page.

```

580 \newif\ifAMCsz@logged\AMCsz@loggedfalse
581 \newcommand{\AMCscoreZone}[1]{%
582   \ifAMC@ensemble%
583     \ifAMCformulaire@dedans%
584       \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
585     \else%
586       \AMC@tracebox{1}{scorequestion::\the\AMCid@quest,-1}{#1}%
587     \fi%
588   \else%
589     \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
590   \fi%
591   \ifAMCsz@logged\else%
592     \AMCmessage{VAR:scorezones=1}%
593     \global\AMCsz@loggedtrue%
594   \fi%
595 }

```

#### 4.8.5 Binary boxes

The package prints on each page some boxes that code (like binary digits) student sheet number, page number and a check number, so as to be read easily from scans after exam.

**\AMCid@checkmax** The check number is just decreased each page. Its maximum value is `\AMCid@checkmax`.  
**\AMC@NCBetud** The number of binary digits used to print student sheet number, page and check number are `\AMC@NCBetud`, `\AMC@NCBpage` and `\AMC@NCBcheck`. The number of the first page is `\AMC@NCBcheck`.  
**\AMC@premierecopie** The length of zone reserved for binary boxes is `\AMC@CBtaille`.

```

596 \def\AMCid@checkmax{60}
597 \def\AMC@NCBetud{12}
598 \def\AMC@NCBpage{6}
599 \def\AMC@NCBcheck{6}
600 \newlength{\AMC@CBtaille}\setlength{\AMC@CBtaille}{5cm}
601 \def\AMC@premierecopie{1}

```

**\AMC@binaryCode** The command `\AMC@binaryCode{<options>}{<n>}` prints boxes to represent the number `<n>` in its binary form. Options from `<options>` include:

`ndigits=<ndigits>` for the number of digits to be shown.

`id=<id>` for an ID of the number role (1 for the student number, 2 for the page number, 3 for the checking value).

`hsep=<hsep>` for the space between boxes.

`style=<style>` for some box style options.



\AMCbin@one and \AMCbin@zero print individual digit-boxes.

For example, \AMC@binaryCode{ndigits=12}{367} shows  $367 = 000101101111_2$  using 12 boxes:



```

602 \newtoks\AMCbin@sequence
603 \newcount\AMCbin@number
604 \newcount\AMCbin@digit
605 \newcount\AMCbin@id
606 \newcount\AMCbin@did
607 \newcount\AMCbin@ndigits
608 \newdimen\AMCbin@hsep
609 \define@key{AMCbin}{ndigits}{\AMCbin@ndigits=#1}
610 \define@key{AMCbin}{id}{\AMCbin@id=#1}
611 \define@key{AMCbin}{hsep}{\AMCbin@hsep=#1}
612 \define@key{AMCbin}{style}[]{\def\AMCbin@style{#1}}
613 \def\AMCbin@one{%
614   \ifnum\AMCbin@did>\z@%
615     \hspace{\AMCbin@hsep}%
616     \fi%
617     \advance\AMCbin@did\@ne%
618     \ifnum\AMCbin@id>0%
619       \AMC@answerBox@{\AMC@checkedbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
620       \else%
621       \AMC@answerBox@{\AMC@checkedbox}{1}{}%
622       \fi}
623 \def\AMCbin@zero{%
624   \ifnum\AMCbin@did>\z@%
625     \hspace{\AMCbin@hsep}%
626     \fi%
627     \advance\AMCbin@did\@ne%
628     \ifnum\AMCbin@id>0%
629       \AMC@answerBox@{\AMC@checkedbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
630       \else%
631       \AMC@answerBox@{\AMC@checkedbox}{1}{}%
632       \fi}
633 \newcommand{\AMC@binaryCode}[2]{%
634   \setkeys{AMCbin}{ndigits=1,hsep=0pt,style}\setkeys{AMCbin}{#1}%
635   \AMCbin@did=\z@%
636   {\AMCboxDimensions{shape=square,size=.32cm,down=0pt,rule=.2pt,cross=false}\expandafter\AMCboxDimensions\
637   \AMCbin@digit=\z@%
638   \loop%
639   \ifnum\AMCbin@number>\z@%
640   \advance\AMCbin@digit\@ne%
641   \ifodd\AMCbin@number\AMCbin@sequence=\expandafter{\expandafter\AMCbin@one\the\AMCbin@sequence}%
642   \else\AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\fi%
643   \divide\AMCbin@number\tw@%
644   \repeat%
645   \loop\relax%
646   \ifnum\AMCbin@digit<\AMCbin@ndigits\advance\AMCbin@digit\@ne%
647   \AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\repeat%
648   \the\AMCbin@sequence%
649   \ifnum\AMCbin@digit>\AMCbin@ndigits\PackageError{automultiplechoice}{Too low AMC@NCB value (got \the\AMCbin@ndigits)}{}

```

```
650 }}
```

The commands `\AMCbin@begin` and `\AMC@binaryBoxes` are now unused and are defined for backward compatibility.

```
651 \def\AMCbin@begin#1{\setkeys{AMCbin}{id=#1}}
652 \newcommand{\AMC@binaryBoxes}[2][1]{%
653 \AMC@binaryCode{ndigits=#1}{#2}%
654 }
```

## 4.9 Checking Environment

`\AMCcurrentenv` Sets the current environment as document.

```
655 \def\AMCcurrentenv{document}
```

`\AMCif@env` Checks for the current environment.

```
656 \def\AMCif@env#1{
657   \def\AMC@tempenv{#1}%
658   \ifx\AMC@tempenv\AMCcurrentenv
659     \expandafter\@firstoftwo
660   \else
661     \expandafter\@secondoftwo
662   \fi
663 }
```

## 4.10 Handling groups of questions

The package allows to handle groups of questions, so as to be able to shuffle them before printing them to the sheets.

`\nouveaugroupe` Command `\nouveaugroupe{<group-name>}{<n>}` creates a new (empty) group with name `<group-name>` (argument `<n>` is present only for compatibility reasons and is ignored). Command `\element{<group-name>}{<text>}` adds to group `<group-name>` a new element that contains `<text>`. `<text>` can be a `question` environment, ore two successive `questions` to be kept together, or anything else. Calling command `\nouveaugroupe` is not compulsory, as `\element` calls it if necessary.

```
664 \newcount\AMCtok@k
665 \newcount\AMCtok@max
666 \newcount\AMCtok@size
667 \newcommand{\nouveaugroupe}[2]{%
668   \expandafter\ifx\csname #1@k\endcsname\relax%
669     \expandafter\newcount\csname #1@k\endcsname%
670     \expandafter\newcount\csname AMC#1@j\endcsname%
671     \csname #1@k\endcsname=\z@\relax%
672     \csname AMC#1@j\endcsname=\z@\relax%
673     \setgroupmode{#1}{\AMCdefault@groupmode}%
674   \fi%
675 }
676 \newcommand\AMC@prepare@element[1]{%
677   \nouveaugroupe{#1}{}%
678   \global\advance\csname #1@k\endcsname\@ne\relax%
679   \AMCtok@k=\csname #1@k\endcsname%
680   \expandafter\ifx\csname #1@romannumeral\AMCtok@k\endcsname\relax%
```

```

681 \expandafter\newtoks\csname #1@\romannumeral\AMCtok@k\endcsname\fi%
682 }
683 \newcommand{\element}[2]{%
684 \AMC@prepare@element{#1}%
685 \global\csname #1@\romannumeral\AMCtok@k\endcsname={#2}%
686 }

```

`\setgroupmode` Command `\setgroupmode{<group-name>}{<mode>}` sets the group mode to `<mode>` for group `<group-name>`. This mode setup the behaviour of `\insertgroup` and `\copygroup` for this group:

1. With mode `fixed`, group's elements will be taken from the beginning.
2. With mode `cyclic`, the elements will be taken from the group following the last call group's use, recycling if necessary.
3. Mode `withreplacement` is the same as `fixed`, but the group is shuffled before each use.
4. Mode `withoutreplacement` is like `cyclic`, adding some shuffling when coming back to the beginning of the group.

The command `\setdefaultgroupmode{<mode>}` sets the group mode to be used for the following created groups (a group is created at the first `\element{<group>}` call). When no `\setdefaultgroupmode` is used, `fixed` is the default mode.

```

687 \def\AMCdefault@groupmode{fixed}
688 \newcommand{\setdefaultgroupmode}[1]{\def\AMCdefault@groupmode{#1}}
689 \newcommand{\setgroupmode}[2]{%
690 \expandafter\ifx\csname AMCgrouppre@#2\endcsname\relax%
691 \PackageError{automultiplechoice}{Unknown group mode for #1 : #2}%
692 {You asked to set group '#1' mode to '#2',
693 but '#2' is not a valid group mode}%
694 \else%
695 \expandafter\global\expandafter\def\csname AMC#1@mode\endcsname{#2}%
696 \fi%
697 }

```

The functions `\AMCgrouppre@xxx{<group-name>}{<n>}{<i>}` are called before using `<n>` elements from group `<group-name>` starting from index `<i>` (negative value for `<i>` stands for the current value of the group index), either with `\insertgroup` or `\copygroup`.

For mode `fixed`, the group index is set to `<i>`, or 0 if `<i>` is negative (take elements from the beginning).

```

698 \newcommand{\AMCgrouppre@fixed}[3]{%
699 \ifnum#3<\z@%
700 \csname AMC#1@j\endcsname=\z@%
701 \else%
702 \csname AMC#1@j\endcsname=#3%
703 \fi%
704 }

```

For mode `withreplacement`, the group is shuffled and the group index is set to `<i>` or 0 (take elements from the beginning) if negative.

```

705 \newcommand{\AMCgrouppre@withreplacement}[3]{%
706 \ifnum#3<\z@%
707 \csname AMC#1@j\endcsname=\z@%
708 \else%

```

```

709 \csname AMC#1@j\endcsname=#3%
710 \fi%
711 \shufflegroup{#1}%
712 }

```

For mode **withoutreplacement**, the group index is set to  $\langle i \rangle$ , or left unchanged if  $\langle i \rangle$  is negative. If there is not enough elements left in the group, the elements before the index and the elements after the index are shuffled.

```

713 \newcount\AMC@imax
714 \newcommand{\AMCgrouppre@withoutreplacement}[3]{%
715 \ifnum#3<\z@%
716 \else%
717 \csname AMC#1@j\endcsname=#3%
718 \fi%
719 \ifnum\AMCtok@ik=\AMCloop@k%
720 \AMCtok@ik=\z@%
721 \fi%
722 \ifnum\AMCtok@ik=\z@%
723 \shufflegroup{#1}%
724 \else%
725 \AMC@imax=\AMCloop@k%
726 \advance\AMC@imax -#2\relax%
727 \ifnum\AMCtok@ik>\AMC@imax%
728 \shufflegroupslice{#1}{\@ne}{\AMCtok@ik}%
729 \ifnum\AMCtok@ik<\AMCloop@k%
730 \advance\AMCtok@ik\@ne%
731 \shufflegroupslice{#1}{\AMCtok@ik}{\AMCloop@k}%
732 \fi%
733 \fi%
734 \fi%
735 }

```

For mode **cyclic**, nothing has to be done, except setting the group index if non-negative.

```

736 \newcommand{\AMCgrouppre@cyclic}[3]{%
737 \ifnum#3<\z@%
738 \else%
739 \csname AMC#1@j\endcsname=#3%
740 \fi%
741 }

```

The function `\AMCgroup@pre{<mode>}{<group-name>}{<n>}{<i>}` calls the right `\AMCgrouppre@xxx` command.

```

742 \newcommand{\AMCgroup@pre}[4]{%
743 \csname AMCgrouppre@#1\endcsname{#2}{#3}{#4}%
744 }

```

`\shufflegroup` Command `\shufflegroup{<group-name>}` shuffles the elements of group  $\langle group-name \rangle$ , and `\insertgroup` `\shufflegroupslice{<group-name>}{<a>}{<b>}` shuffles elements  $\langle a \rangle$  to  $\langle b \rangle$  from group  $\langle group-name \rangle$ . `\insertgroupfrom` It can be called at each student sheet in order to get different student sheets and avoid cheating.

Command `\insertgroup[<n>]{<groupname>}` inserts all the elements of group  $\langle groupname \rangle$ , or only the first  $\langle n \rangle$  elements if  $\langle n \rangle$  is given. `\insertgroupfrom[<n>]{<groupname>}{<i>}` inserts all the elements of group  $\langle groupname \rangle$  starting from index  $\langle i \rangle$  (the index of the first element is 0), or only the first  $\langle n \rangle$  elements if  $\langle n \rangle$  is given.

```

745 \newcommand{\shufflegroup}[1]{%

```

```

746 \ifAMC@shuffleG{\AMC@shuffletoks{\number\csname #1@k\endcsname}{#1@}}\fi%
747 }
748 \newcommand{\shufflegroupslice}[3]{%
749 \ifAMC@shuffleG{\AMC@shuffletoks[#2]{#3}{#1@}}\fi%
750 }
751 \newcount\AMCtok@ik
752 \newcount\AMCloop@k
753 \newcommand{\AMCgrouploop@prep}[3]{%
754 \AMCtok@size=#1\relax%
755 \ifAMC@fullGroups\AMCtok@size=\m@ne\fi%
756 \ifnum\AMCtok@size<\z@%
757 \AMCtok@size=\csname #2@k\endcsname%
758 \fi%
759 \AMCtok@ik=\csname AMC#2@j\endcsname%
760 \AMCloop@k=\csname #2@k\endcsname%
761 \expandafter\ifx\csname AMC#2@mode\endcsname\relax%
762 \PackageError{automultiplechoice}{No group mode for #2}%
763 {No mode has been defined for group '#2'. This should not occur...}%
764 \fi%
765 \AMCgroup@pre{\csname AMC#2@mode\endcsname}{#2}{\the\AMCtok@size}{#3}%
766 }
767 \newcommand{\AMCgrouploop@next}[1]{%
768 \global\advance\csname AMC#1@j\endcsname\@ne\relax%
769 \expandafter\ifnum\csname AMC#1@j\endcsname>\AMCloop@k\relax%
770 \global\csname AMC#1@j\endcsname=\@ne%
771 \fi%
772 \AMCtok@ik=\csname AMC#1@j\endcsname%
773 \advance\AMCtok@size\m@ne%
774 }
775 \newcommand{\insertgroupfrom}[3][-1]{%
776 \ifnum#1=0%
777 \else%
778 \AMCgrouploop@prep{#1}{#2}{#3}%
779 {\loop%
780 \AMCgrouploop@next{#2}%
781 {\the\csname #2@romannumeral\AMCtok@ik\endcsname}%
782 \ifnum\AMCtok@size>\z@\repeat}%
783 \fi%
784 }
785 \newcommand{\insertgroup}[2][-1]{%
786 \insertgroupfrom[#1]{#2}{-1}%
787 }

```

`\cleargroup` The commands `\cleargroup` and `\copygroup` can also be used to make more complex questions combinations in the exams, allowing for example to ask the package to shuffle 3 questions taken at random from group `groupa` and 5 questions taken at random from group `groupb`.

`\cleargroup{⟨group⟩}` clears the group `⟨group⟩`, ereasing all of its elements.

`\copygroup[⟨n⟩]{⟨from⟩}{⟨to⟩}` copies `⟨n⟩` elements from group `⟨from⟩` to group `⟨to⟩`. If optional parameter `⟨n⟩` is not given, all the questions from group `⟨from⟩` are copied. `\copygroupfrom[⟨n⟩]{⟨from⟩}{⟨to⟩}{⟨i⟩}` copies `⟨n⟩` elements from group `⟨from⟩` to group `⟨to⟩`, starting from element at index `⟨i⟩` (the index of the first element is 0). If optional parameter `⟨n⟩` is not given, all the questions from group `⟨from⟩` are copied.

See section 3.4 for an illustration for these commands.

```

788 \newcommand{\cleargroup}[1]{%
789   \nouveaugroupe{#1}{}%
790   \csname #1@k\endcsname=\z@\relax%
791   \csname AMC#1@j\endcsname=\z@\relax%
792 }
793 \newcommand{\copygroupfrom}[4][-1]{%
794   \ifnum#1=0%
795     \else%
796       \AMCgrouploop@prep{#1}{#2}{#4}%
797       {\loop%
798         \AMCgrouploop@next{#2}%
799         \AMC@prepare@element{#3}%
800         \global\csname #3@\romannumeral\AMCtok@k\endcsname=\csname #2@\romannumeral\AMCtok@ik\endcsname%
801         \ifnum\AMCtok@size>\z@\repeat}%
802       \fi%
803 }
804 \newcommand{\copygroup}[3][-1]{%
805   \copygroupfrom{#1}{#2}{#3}{-1}%
806 }

```

## 4.11 Questions

To manage multiple choice questions, first set some counters and token registers to handle answers. Token registers `\reponse@i`, `\reponse@ii` and so on will be used for answers – we restrict the number of answers of a single questions to `\AMCload@counter = 199`.

```

807 \newcount\AMCrep@count
808 \AMCload@counter=199
809 \@whilenum\AMCload@counter>0\do{%
810   \expandafter\newtoks\csname reponse@\romannumeral\AMCload@counter\endcsname%
811   \advance\AMCload@counter@m@ne%
812 }

```

`\AMCload@reponse` Command `\AMCload@reponse{<n>}{<text>}` will be used to add answer number `<n>` with text `<text>` (`<text>` will include the box to be ticked and all the layout commands) to the set of answers (in a token register `\reponse@xxx` – counter `\AMCload@counter` keeps track of the number of answers), in order to shuffle them when all answers will be loaded.

When answers are not to be shuffled, command `\AMCrien@deux{<n>}{<text>}` will be used instead, only printing `<text>`.

```

813 \newcommand\AMCload@reponse[2]{%
814   \global\advance\AMCload@counter\@ne\relax%
815   \global\csname reponse@\romannumeral\AMCload@counter\endcsname%
816   =\expandafter{\expandafter\AMCrep@count\expandafter=#2 #1}%
817 }
818 \newcommand\AMCrien@deux[2]{#1}

```

`\shuffle@it` After loading all answers, commands `\shuffle@it` will be used to shuffle them, and `\AMCdump@reponses` to print them.

```

819 \def\shuffle@it{\AMC@shuffletoks{\number\AMCload@counter}{reponse@}}
820 \newcount\AMCnum@questions
821 \newcommand\AMCdump@reponses{%
822   \global\AMCnum@questions=\AMCload@counter%
823   \@whilenum\AMCload@counter>0\do{%

```

```

824 \the\csname reponse@\romannumeral\AMCload@counter\endcsname%
825 \advance\AMCload@counter\m@ne}}

```

#### 4.11.1 Managing answers

`\lastchoices` Command `\AMCrep@init{<mode>}` is called for each question before reading answers. `<mode>` `\AMCrep@init` is `r` for suffled answers, and `o` if answers are not to be shuffled. It sets the number of answers `\AMC@fin@rep` counter to zero, and calls `\AMCrep@o` or `\AMCrep@r` depending on `<mode>`. These commands sets `\AMCload@@reponse` and `\AMCrep@fini` that will be called for each answer and after the last answer respectively, depending on `<mode>`:

- If `<mode>=r`, `\AMCload@@reponse` is `\AMCload@reponse` (loads answer to token register) and `\AMCrep@fini` calls `\shuffle@it` and `\AMCdump@reponses`;
- If `<mode>=o`, `\AMCload@@reponse` is `\AMCrien@deux` (prints answer directly) and `\AMCrep@fini` does nothing.

Command `\lastchoices` is called before giving answers that are to be printed at the end (even when shuffling answers). It closes the answers list calling `\AMCrep@fini` and opens another one in ordered mode. Note that it also saves the value of `\AMCrep@count`, which is the number of the current answer among all answers given in the subject source for the current question.

Command `\AMC@fin@rep` is to be called after the last answer: it adds a “None of these answers are correct.” answer if necessary (package option `completemulti`) with answer number zero, and calls `\AMCrep@fini`.

```

826 \newcommand\AMCrep@init[1]{%
827   \ifAMC@ordre\AMCrep@o\else%
828     \csname AMCrep@#1\endcsname\fi\AMCload@counter=\z@}
829 \newcommand\AMCrep@o{%
830   \def\AMCload@@reponse{\AMCrien@deux}\def\AMCrep@fini{}}
831 \newcommand\AMCrep@r{%
832   \def\AMCload@@reponse{\AMCload@reponse}%
833   \def\AMCrep@fini{\shuffle@it\AMCdump@reponses}}
834 \newcount\AMCrep@count
835 \newcommand\lastchoices{%
836   \AMCrep@count=\AMCrep@count%
837   \AMCrep@fini\AMCrep@init{o}%
838   \AMCrep@count=\AMCrep@count}
839 \newcommand\@aucune{\emph{\AMC@loc@none}}
840 \newcommand\AMC@fin@rep{%
841   \ifAMCcomplete@multi\ifAMCtype@multi%
842     \lastchoices\AMCrep@count=-1%
843     \ifAMC@bonne\wrongchoice{\@aucune}\else%
844       \ifAMC@postcorrect\wrongchoice{\@aucune}\else\correctchoice{\@aucune}\fi%
845     \fi\fi\fi\AMCrep@fini}

```

#### 4.11.2 Separate answer sheet

This package needs some memory to print questions/answers boxes again on a separate answer sheet.

`\AMCformQuestion` First define commands that will announce questions and answers on the separate answer sheet  
`\AMCformAnswer` (these commands can be modified by the user): `\AMCformQuestion{<number>}` is responsible

for announcing question, and `\AMCformAnswer{<box>}` is responsible for printing the box to be ticked, given as argument `<box>`.

Commands `\AMCformQuestionA` and `\AMCformAnswerA` set up counter `\AMC@ncase` value before calling their counterparts.

```

846 \def\AMCformBeforeQuestion{\vspace{\AMCformVSpace}\par}
847 \def\AMCformAfterQuestion{\ifAMC@asqbloc\egroup\fi}
848 \def\AMCformQuestion#1{\AMC@loc@qf{#1}}
849 \def\AMCformQuestionN{\AMCformQuestion{\AMC@qaff}}
850 \def\AMCformQuestionA{%
851   \setcounter{AMC@ncase}{0}%
852   \AMCformBeforeQuestion%
853   \ifAMC@asqbloc\vbox\bgroup\fi%
854   \ifx\@empty\AMC@sza@callout\@empty\else%
855     \csname\AMC@sza@callout\endcsname%
856   \fi%
857   \AMCformQuestionN%
858   \ifx\@empty\AMC@sza@callin\@empty\else%
859     \csname\AMC@sza@callin\endcsname%
860   \fi%
861 }
862 \def\AMCformAnswer#1{\hspace{\AMCformHSpace} #1}
863 \def\AMCformAnswerA#1{\addtocounter{AMC@ncase}{1}\AMCformAnswer{#1}}

```

`\AMC@mem@add@ifneeded` These are commands to manage memory for separate answer sheet. `\AMC@mem@add@ifneeded{<code>}`

`\AMCformBegin` adds `<code>` to this memory. `\AMC@mem@answer{<code>}` adds to memory answer code `<code>`, and

`\AMCform` `\AMC@mem@openQuestion` adds to memory question code to announce current question.

`\AMCformS` The command `\AMCformBegin` defines the beginning of the separate answer sheet for the current student sheet, and `\AMCform` prints the whole memory: questions and answers boxes.

`\AMCformS` is a `\AMCform` variant that does not clear the list of answer boxes. It can be used to make the same exact subject for all students, displaying the questions before (outside) `onecopy`, so that `onecopy` contains only the answer sheet.

```

864 \ExplSyntaxOn
865
866 \prg_set_conditional:Nnn \amc_if_separate_question: { p , T } {
867   \ifAMC@ensemble
868     \ifAMC@zoneformulaire
869       \prg_return_false:
870     \else
871       \prg_return_true:
872     \fi
873   \else
874     \prg_return_false:
875   \fi
876 }
877 \cs_new_eq:NN \AMC@ifseparate@question \amc_if_separate_question:T
878
879 \int_new:N \amc_memory_elts_count
880
881 \cs_new:Nn \amc_clear_memory: { \int_gzero:N \amc_memory_elts_count }
882 \cs_new_eq:NN \AMC@mem@clear \amc_clear_memory:
883

```



```

884 \cs_new:Npn \amc_memory_elt_i:n #1 {
885   amc_memory_elts_ \int_to_alph:n { #1 }
886 }
887 \cs_new:Nn \amc_memory_current_elt: {
888   \amc_memory_elt_i:n \amc_memory_elts_count
889 }
890 \cs_new:Npn \amc_memory_vars_i:n #1 {
891   amc_memory_vars_ \int_to_alph:n { #1 }
892 }
893 \cs_new:Nn \amc_memory_current_vars: {
894   \amc_memory_vars_i:n \amc_memory_elts_count
895 }
896
897 \cs_new:Nn \amc_add_memory_elt: {
898   \int_gincr:N \amc_memory_elts_count
899   \tl_gclear_new:c { \amc_memory_current_elt: }
900   \tl_gclear_new:c { \amc_memory_current_vars: }
901 }
902 \cs_new_eq:NN \AMC@mem@next \amc_add_memory_elt:
903
904 \cs_new:Npn \amc_add_to_memory:n #1 {
905   \tl_gput_right:cn { \amc_memory_current_elt: } { #1 }
906 }
907 \cs_new_eq:NN \AMC@mem@add \amc_add_to_memory:n
908
909 \cs_new:Npn \amc_add_to_vars:n #1 {
910   \tl_gput_right:cn { \amc_memory_current_vars: } { #1 }
911 }
912 \cs_new_eq:NN \AMC@mem@addvar \amc_add_to_vars:n
913
914 \cs_new:Npn \amc_add_qidaffname:nnn #1#2#3 {
915   \amc_add_to_vars:n {\AMCid@quest=#1\setcounter{AMCquestionaff}{#2}%
916     \global\def\AMCid@name{#3}}
917 }
918 \cs_generate_variant:Nn \amc_add_qidaffname:nnn { xxx }
919 \cs_new_eq:NN \AMC@mem@qidaffname \amc_add_qidaffname:xxx
920
921 \cs_new:Npn \amc_mem_elt_cat:n #1 {
922   \amc_add_to_vars:n { \def\AMCmem@elt@cat{ #1 } }
923 }
924 \cs_generate_variant:Nn \amc_mem_elt_cat:n { x }
925 \cs_new_eq:NN \AMC@mem@category \amc_mem_elt_cat:x
926
927 \cs_new:Npn \amc_add_aid:n #1 {
928   \amc_add_to_memory:n {\AMCrep@count=#1}
929 }
930 \cs_generate_variant:Nn \amc_add_aid:n { x }
931 \cs_new_eq:NN \AMC@mem@aid \amc_add_aid:x
932
933 \cs_new:Npn \amc_if_category_is_p:n #1 {
934   \str_if_eq_p:on { \AMCmem@elt@cat } { #1 }
935 }
936 \cs_new:Npn \amc_use_memory:n #1 {

```

```

937 \int_step_inline:nnnn { 1 } { 1 } \amc_memory_elts_count {
938   \def\AMCmem@elt@cat{ plain }
939   \tl_use:c { \amc_memory_vars_i:n { ##1 } }
940   \bool_if:nTF { #1 } {
941     \tl_use:c { \amc_memory_elt_i:n { ##1 } }
942   } { }
943 }
944 }
945 \cs_new:Nn \amc_use_memory: { \amc_use_memory:n { \c_true_bool } }
946 \cs_new_eq:NN \AMC@mem@show \amc_use_memory:
947 \cs_new_eq:NN \AMC@mem@show@filter \amc_use_memory:n
948 \cs_new_eq:NN \AMC@category \amc_if_category_is_p:n
949
950 \ExplSyntaxOff
951 \newcommand\AMC@mem@add@ifneeded[1]{%
952   \AMC@if@separate@question{%
953     \AMC@mem@add{#1}%
954   }%
955 }
956 \newcommand\AMC@mem@addsingle@ifneeded[2]{%
957   \AMC@if@separate@question{%
958     \AMC@mem@next%
959     \AMC@mem@category{#2}%
960     \AMC@mem@add{#1}%
961   }%
962 }
963 \newcommand\AMC@mem@answer[1]{%
964   \addtocounter{AMC@ncase}{1}%
965   \AMC@if@separate@question{%
966     \AMC@mem@aid{\the\AMC@rep@count}%
967     \AMC@mem@add{\AMCformAnswerA{#1}}%
968   }%
969 }
970 \newcommand\AMC@mem@openQuestion{%
971   \AMC@if@separate@question{%
972     \AMC@mem@next%
973     \AMC@mem@qidaffname{\the\AMC@id@quest}{\arabic{AMC@questionaff}}{\AMC@id@name}%
974     \AMC@mem@add{\AMCformQuestionA}%
975   }%
976 }
977 \def\AMCformBegin{%
978   \AMC@zoneformulairetrue\setcounter{section}{0}%
979   \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageFull}\fi\fi%
980 }
981 \newcommand\AMCform{%
982   \ifAMC@ensemble\AMCformulaire@dedanstrue%
983     \AMC@mem@show%
984   \fi}
985 \newcommand\AMCformFilter[1]{%
986   \ifAMC@ensemble\AMCformulaire@dedanstrue%
987     \AMC@mem@show@filter{#1}%
988   \fi}
989 \newif\ifAMC@keepmemory

```

```

990 \newcommand\AMCformS{%
991   \ifAMC@ensemble\AMCformulaire@dedanstrue%
992   \AMCmessage{BR=0}\AMC@mem@show%
993   \global\AMC@keepmemorytrue%
994   \fi}

```

`\AMCsection` The `\AMCsection` and `\AMCsubsection` commands issue their standard counterparts (`\section` and `\subsection` with the same argument, both in the subject *and* in the separate answer sheet.

```

995 \newcommand{\AMCsectionNumbered}[1]{%
996   \section{#1}\AMC@mem@addsingle@ifneeded{\section{#1}}{section}}
997 \newcommand{\AMCsubsectionNumbered}[1]{%
998   \subsection{#1}\AMC@mem@addsingle@ifneeded{\subsection{#1}}{subsection}}
999 \newcommand{\AMCsectionStar}[1]{%
1000   \section*{#1}\AMC@mem@addsingle@ifneeded{\section*{#1}}{section}}
1001 \newcommand{\AMCsubsectionStar}[1]{%
1002   \subsection*{#1}\AMC@mem@addsingle@ifneeded{\subsection*{#1}}{subsection}}
1003 \def\AMCsection{\@ifstar\AMCsectionStar\AMCsectionNumbered}
1004 \def\AMCsubsection{\@ifstar\AMCsubsectionStar\AMCsubsectionNumbered}

```

### 4.11.3 Formatting answers

`choices` (*env.*) Answers have to be included in an environment `choices` (standard), `choiceshoriz` (answers on one line) or `choicescustom` (user defined) depending on the desired formatting.

`choiceshoriz` (*env.*) Use `\AMCBoxedAnswers` to request all answers to be included in L<sup>A</sup>T<sub>E</sub>X boxes; this can be useful for example when using multicolumn answers formatting.

`tikz-single` (*env.*)

`tikz-multi` (*env.*)

```

1005 \def\AMCBoxedAnswers{\AMC@rbloctrue}
\AMCBoxedAnswers 1006 \newenvironment{choices}[1][r]{%
1007   \AMCrep@count=\z@\def\une@rep{\AMCrep@itemize}%
1008   \ifAMC@rbloc\def\une@rep{\AMCrep@bloc}%
1009   \else\begin{itemize}\setlength{\itemsep}{\AMCinterIrep}\fi%
1010   \AMCrep@init{#1}}%
1011   {\AMC@fin@rep\ifAMC@rbloc\else\end{itemize}\fi}
1012 \newenvironment{choiceshoriz}[1][r]{%
1013   \AMCrep@count=\z@\def\une@rep{\AMCrep@ligne}\AMCrep@init{#1}%
1014   \par\begin{center}}%
1015   {\AMC@fin@rep\end{center}}
1016 \newenvironment{choicescustom}[1][r]{%
1017   \AMCrep@count=\z@\def\une@rep{\AMCrep@perso}\AMCrep@init{#1}%
1018   \AMCbeginAnswer\ignorespaces}%
1019   {\AMC@fin@rep\AMCendAnswer}
1020 \newenvironment{tikz-single}[1][r]{
1021   \AMCrep@count=\z@\def\une@rep{\AMCrep@tikz}\AMCrep@init{#1}%
1022   \begin{tikzpicture}}{\AMC@fin@rep\end{tikzpicture}}
1023 \newenvironment{tikz-multi}[1][r]{
1024   \AMCrep@count=\z@\def\une@rep{\AMCrep@tikzmat}\AMCrep@init{#1}%
1025   \begin{tikzpicture}[remember picture]}{\AMC@fin@rep\end{tikzpicture}}

```

`\AMCrep@bloc` For each of these styles, a corresponding `\AMCrep@xxx{\langle box \rangle}{\langle text \rangle}` is defined, which will format the answer with a box given in  $\langle box \rangle$  and text  $\langle text \rangle$ . `\AMCrep@bloc` is also defined and used in standard formatting when the user wants to put answers inside a L<sup>A</sup>T<sub>E</sub>X box.

```

\AMCrep@tikz 1026 \newcommand\AMCrep@bloc[2]{\AMC@mem@answer{#1}%
\AMCrep@tikzmat 1027   \par%
\AMCrep@itemize
\AMCrep@ligne
\AMCrep@perso

```

```

1028 \ifAMC@textPos\ vbox\bgroup\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\hbox{
1029 \noindent\begin{minipage}{\linewidth}%
1030 \begin{itemize}\item{#1} #2\end{itemize}\end{minipage}%
1031 \ifAMC@textPos\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\egroup\AMC@tra
1032 \vspace{\AMCinterBrep}}
1033 \newcommand\AMCrep@tikz[5]{\AMC@mem@answer{#1}\node[#4] (lab\thecsvrow) at (#3) {#2} node[#5] (box\thecsvrow)
1034 \newcommand\AMCrep@tikzmat[5]{\AMC@mem@answer{#1}\node[#5] (box\thecsvrow) at (#3) {#1} node[#4] (lab\thecsvrow)
1035 \newcommand\AMCrep@itemize[2]{\AMC@mem@answer{#1}\item{#1} #2}
1036 \newlength\AMChorizAnswerSep
1037 \setlength\AMChorizAnswerSep{3em plus 4em}
1038 \newlength\AMChorizBoxSep
1039 \setlength\AMChorizBoxSep{1em}
1040 \newcommand\AMCrep@ligne[2]{\AMC@mem@answer{#1}%
1041 \ifAMC@textPos%
1042 \mbox{\AMC@tracebox{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}{#1}\hspace*\AMChorizBoxSep}%
1043 \else%
1044 \mbox{#1\hspace*\AMChorizBoxSep}{#2}%
1045 \fi\hspace*\AMChorizAnswerSep}}
1046 \newcommand\AMCrep@perso[2]{\AMC@mem@answer{#1}\AMCanswer{#1}{#2}}

```

`\AMCbeginAnswer` The custom style will use user-defined commands to format answers: `\AMCbeginAnswer` is called once before answers, `\AMCanswer{<box>}{<text>}` is called for each answer (`<box>` being the box to be ticked and `<text>` the text associated with the proposed answer), and `\AMCendAnswer` is called after all answers.

```

1047 \def\AMCbeginAnswer{}
1048 \def\AMCanswer#1#2{#1 #2}
1049 \def\AMCendAnswer{}

```

`\answer` The commands `\correctchoice` and `\wrongchoice` are used inside `choices`-like environments to give the proposed answers and specify if they are to be ticked by the students or not.

```

\wrongchoice
1050 \newcommand{\correctchoice}[2][]{\global\advance\AMCrep@count\@ne\relax%
1051 \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%
1052 \global\AMCune@bonnettrue%
1053 \AMCload@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkedbox}%
1054 \else\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}\ignorespaces}
1055 \newcommand{\wrongchoice}[2][]{\global\advance\AMCrep@count\@ne\relax%
1056 \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:M}\fi%
1057 \AMCload@reponse{\une@rep{\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}%
1058 \ignorespaces}

```

#### 4.11.4 Score zones

`\AMCscoreZone` The position of the scores on the annotated answer sheets can be defined in the L<sup>A</sup>T<sub>E</sub>X source file using `\AMCsetScoreZone{<options>}` (or `\AMCsetScoreZoneAnswerSheet{<options>}` for the answer sheets when the separate answer sheet option is used).

First begin with some helpers: `\AMCemptybox{<width>}{<height>}{<depth>}` draws an empty box with specified dimensions, and `\AMCmarginNote{<note>}` (code from one of [sgmoye's](http://tex.stackexchange.com) comments on [tex.stackexchange.com](http://tex.stackexchange.com)) prints a marginal note in the left or right margin, depending on current the position (usefull in `multicols` environment).

```

1059 \newcommand{\AMCemptybox}[3]{%
1060 \sbox0{\wd0=#1\ht0=#2\dp0=#3\relax\box0}}
1061 \newlength\AMC@mn@test

```

```

1062 \newlength\AMC@mn@sep\AMC@mn@sep=4mm
1063 \newlength\AMC@mn@leftmargin
1064 \newlength\AMC@mn@rightmargin
1065 \newcommand\AMCmarginNote[1]{%
1066   \begin{tikzpicture}[remember picture,overlay]%
1067     \coordinate (here) at (0,0);%
1068     \pgfextractx{\AMC@mn@test}{\pgfpointdiff{\pgfpointorigin}%
1069       {\pgfpointanchor{current page}{center}}}%
1070     \ifodd\thepage%
1071       \AMC@mn@leftmargin=\oddsidemargin%
1072       \AMC@mn@rightmargin=\evensidemargin%
1073     \else
1074       \AMC@mn@leftmargin=\evensidemargin%
1075       \AMC@mn@rightmargin=\oddsidemargin%
1076     \fi
1077     \ifdim\AMC@mn@test < 1cm%
1078       \draw (current page.east |- here)+(-\AMC@mn@rightmargin-1in+\AMC@mn@sep,0pt) node[anchor=text,align=right]{%
1079         \AMCmarginNote{#1}}%
1080     \else%
1081       \draw (current page.west |- here)+(0cm,0pt) node[anchor=text,align=right,text width=\AMC@mn@leftmargin]{%
1082         \AMCmarginNote{#1}}%
1083     \fi
1084 \end{tikzpicture}%
1085 }

```

Define now different ways to place the score zone:

`none` nowhere

`question` right after the question heading

`margin` in the margin, using `marginpar` (this does not work with `multicols` environment)

`margins` in the left or right margin, depending on the current position (needs `tikz` package)

```

1084 \newcommand{\AMC@sz@box}{\AMCemptybox{\AMC@sz@width}{\AMC@sz@height}{\AMC@sz@depth}}
1085 %
1086 \newcommand{\AMC@sz@callin@question}{\AMCscoreZone{\AMC@sz@box}}
1087 %
1088 \newcommand{\AMC@sz@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sz@box}}}
1089 %
1090 \newcommand{\AMC@sz@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper margins}}
1091 \newcommand{\AMC@sz@callout@margins}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}

```

Let us now set up options handling.

```

1092 \newlength\AMC@sz@width
1093 \newlength\AMC@sz@height
1094 \newlength\AMC@sz@depth
1095 \def\AMC@sz@callout{}
1096 \def\AMC@sz@callin{}
1097 \define@key{AMCsz}{width}{\AMC@sz@width=#1}
1098 \define@key{AMCsz}{height}{\AMC@sz@height=#1}
1099 \define@key{AMCsz}{depth}{\AMC@sz@depth=#1}
1100 \define@key{AMCsz}{calloutside}{\def\AMC@sz@callout{#1}}
1101 \define@key{AMCsz}{callinside}{\def\AMC@sz@callin{#1}}
1102 \define@choicekey{AMCsz}{position}{none,question,margin,margins}{%
1103   \ifcsname AMC@sz@callout@#1\endcsname%

```

```

1104 \def\AMC@sz@callout{AMC@sz@callout@#1}%
1105 \else%
1106 \def\AMC@sz@callout{}%
1107 \fi%
1108 \ifcsname AMC@sz@callin@#1\endcsname%
1109 \def\AMC@sz@callin{AMC@sz@callin@#1}%
1110 \else%
1111 \def\AMC@sz@callin{}%
1112 \fi%
1113 \ifcsname AMC@sz@init@#1\endcsname%
1114 \csname AMC@sz@init@#1\endcsname%
1115 \fi%
1116 }
1117 \newcommand{\AMCsetScoreZone}[1]{\setkeys{AMCsz}{#1}}
1118 \AMCsetScoreZone{width=1.5em,height=1.5ex,depth=.5ex,position=none}

```

And do the same for \AMCsetScoreZoneAnswerSheet...

```

1119 \newcommand{\AMC@sza@box}{\AMCemptybox{\AMC@sza@width}{\AMC@sza@height}{\AMC@sza@depth}}
1120 %
1121 \newcommand{\AMC@sza@init@none}{}
1122 \newcommand{\AMC@sza@callout@none}{}
1123 \newcommand{\AMC@sza@callin@none}{}
1124 %
1125 \newcommand{\AMC@sza@init@question}{}
1126 \newcommand{\AMC@sza@callout@question}{}
1127 \newcommand{\AMC@sza@callin@question}{\AMCscoreZone{\AMC@sza@box}}
1128 %
1129 \newcommand{\AMC@sza@init@margin}{}
1130 \newcommand{\AMC@sza@callout@margin}{\hspace{Opt}\marginpar{\AMCscoreZone{\AMC@sza@box}}}
1131 \newcommand{\AMC@sza@callin@margin}{}
1132 %
1133 \newcommand{\AMC@sza@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper ma}}
1134 \newcommand{\AMC@sza@callout@margins}{\hspace{Opt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}
1135 \newcommand{\AMC@sza@callin@margins}{}
1136 %
1137 \newlength\AMC@sza@width
1138 \newlength\AMC@sza@height
1139 \newlength\AMC@sza@depth
1140 \def\AMC@sza@callout{}
1141 \def\AMC@sza@callin{}
1142 \define@key{AMCsza}{width}{\AMC@sza@width=#1}
1143 \define@key{AMCsza}{height}{\AMC@sza@height=#1}
1144 \define@key{AMCsza}{depth}{\AMC@sza@depth=#1}
1145 \define@key{AMCsza}{calloutsideside}{\def\AMC@sza@callout{#1}}
1146 \define@key{AMCsza}{callinside}{\def\AMC@sza@callin{#1}}
1147 \define@choicekey{AMCsza}{position}{none,question,margin,margins}{%
1148 \ifcsname AMC@sza@callout@#1\endcsname%
1149 \def\AMC@sza@callout{AMC@sza@callout@#1}%
1150 \else%
1151 \def\AMC@sza@callout{}%
1152 \fi%
1153 \ifcsname AMC@sza@callin@#1\endcsname%
1154 \def\AMC@sza@callin{AMC@sza@callin@#1}%
1155 \else%

```

```

1156 \def\AMC@sza@callin{}%
1157 \fi%
1158 \ifcsname AMC@sza@init@#1\endcsname%
1159 \csname AMC@sza@init@#1\endcsname%
1160 \fi%
1161 }
1162 \newcommand{\AMCsetScoreZoneAnswerSheet}[1]{\setkeys{AMC@sza}{#1}}
1163 \AMCsetScoreZoneAnswerSheet{width=1.5em,height=1.5ex,depth=.5ex,position=none}
1164 \newcommand{\AMCnoScoreZone}{\AMCsetScoreZone{position=none}\AMCsetScoreZoneAnswerSheet{position=none}}

```

#### 4.11.5 Formatting questions

`\AMCquestionaff` The counter `\AMCquestionaff` keeps track of the current question number. It can be redefined by the user, for example to print several questions without a number, and then print questions with a number starting at one.

`\AMC@stepQuestion` will increase this counter and `\AMC@qaff` will format the question number out.

```

1165 \newcounter{AMCquestionaff}
1166 \newcommand{\AMCnumero}[1]{\setcounter{AMCquestionaff}{#1}\addtocounter{AMCquestionaff}{-1}}
1167 \AtBeginDocument{%
1168 \ifx\@skiphyperreftrue\@undefined%
1169 \expandafter\newif\csname if@skiphyperref\endcsname%
1170 \fi%
1171 }
1172 \newcommand\AMC@stepQuestion{\ifAMCquestionNumber\@skiphyperreftrue\refstepcounter{AMCquestionaff}\@skip
1173 \newcommand\AMC@qaff{\arabic{AMCquestionaff}}

```

`\AMCbeforeQuestion` The command `\AMCbeforeQuestion` opens a new question. The command `\AMCbeginQuestion{<n>}{<sign>}` will format the question header, where `<n>` is the question number and `<sign>` being `\multiSymbole` in case of a multiple question, and empty in case of a simple one. `\AMCbeforeQuestion`, `\AMCbeginQuestion` and `\multiSymbole` can be user-redefined.

```

1174 \def\AMCbeforeQuestion{\ifAMC@qbloc\else\par\noindent\fi}
1175 \def\AMCbeginQuestion#1#2{\noindent\AMC@loc@q{#1}{#2}%
1176 \ifx\@empty\AMC@sz@callin\@empty\hspace*{1em}\fi%
1177 }
1178 \def\multiSymbole{${\clubsuit}$}

```

`question (env.)` Environment `{question}{<key>}` encloses a simple question (with one and only one correct choice) with associated unique key `<key>` and the proposed answers.

`variable-single (env.)` Environment `{questionmult}{<key>}` is the same for multiple questions (with none, one or several correct choices).

`questionouverte (env.)` Environment `{questionmultx}{<key>}` is the same as `questionmult`, but with no use of `\ouverte@vs` `\multiSymbole`.

Environment `{questionouverte}[<width>]` is used for open questions (that won't be marked automatically!), with width given as an optional argument (defaults to 3 cm).

The command `\AMCexternalQuestion{<id>}{<maxscore>}` allows to declare a question that will be scored outside AMC, with a maximal score `<maxscore>`. When you use this command, you can manage the question number and question text freely (AMC won't handle this).

```

1179 \ifx\question\undefined\else\let\question\undefined\fi
1180 \def\AMCnobloc{\AMC@qblocfalse}
1181 \def\AMCbloc{\AMC@qbloctrue}

```

```

1182 \newcommand\AMCstartWithQuestion[1]{%
1183   \global\def\AMCid@name{#1}\AMC@affecte{#1}{\AMCid@quest}%
1184   \ifAMC@calibration%
1185     \AMCmessage{Q=\the\AMCid@quest}%
1186     \immediate\write\AMC@XYFILE{\string\question{\the\AMCid@quest}{\AMCid@name}}%
1187   \fi%
1188 }
1189 \newcommand\AMCexternalQuestion[2]{%
1190   \AMCstartWithQuestion{#1}%
1191   \ifAMC@calibration%
1192     \AMCmessage{B=MAX=#2}%
1193     \AMCmessage{MULT}%
1194     \AMCmessage{FQ}%
1195   \fi%
1196 }
1197 \newenvironment{question}[2][]{%
1198   \def\AMCcurrentenv{question}%
1199   \AMC@stepQuestion%
1200   \AMCstartWithQuestion{#2}%
1201   \AMCbeforeQuestion%
1202   \ifx\@empty\AMC@sz@callout\@empty\else%
1203     \csname\AMC@sz@callout\endcsname%
1204   \fi%
1205   \AMCtype@multifalse\ifAMC@qbloc\ifAMC@textPos\vbox\bgroup\AMC@tracepos{1}{qtext:#2:\the\AMCid@quest,0}%
1206   \ifAMC@affichekeys\index{\texttt{#2}}\ifAMC@keyline[\texttt{#2}]\newline\fi\fi%
1207   \AMCbeginQuestion{\ifAMC@affichekeys\ifAMC@ensemble\AMC@qaff\ \fi\ifAMC@keyline\else[\texttt{#2}]\fi}%
1208   \ifAMC@calibration\immediate\write\AMC@XYFILE{\string\questionnum{\the\AMCid@etud}{\the\AMCid@quest}{\the\AMCid@name}}%
1209   \ifx\@empty\AMC@sz@callin\@empty\else%
1210     \csname\AMC@sz@callin\endcsname%
1211   \fi%
1212   \AMCformulaire@dedansfalse\setcounter{AMC@ncase}{0}%
1213   \AMC@mem@openQuestion}%
1214 {\ifAMC@qbloc\end{minipage}\ifAMC@textPos\AMC@tracepos{1}{qtext:\AMCid@name:\the\AMCid@quest,0}\egroup\AMC@afterQuestion}%
1215 \newenvironment{questionmult}[1]{%
1216   \AMC@bonnefalse\begin{question}[{\multiSymbole}]{#1}%
1217   \AMCtype@multitruel\ifAMC@calibration%
1218     \AMCmessage{MULT}\fi%
1219 {\end{question}}
1220 \newif\ifSurveySingleAnswer\SurveySingleAnswerfalse
1221 \newenvironment{variable-single}[2]
1222 {\def\AMCbeginQuestion##1##2{
1223 \begin{question}{#1}\scoring{v=#2}\QuestionIndicative\ifSurveySingleAnswer\else\AMCmessage{MULT}\fi
1224 \begin{tikz-single}[o]
1225 {\end{tikz-single}
1226 \end{question}}
1227 \newenvironment{variable-multi}[4]
1228 {\def\AMCbeginQuestion##1##2{
1229 \begin{question}{#1}\scoring{v=#4}\QuestionIndicative\ifSurveySingleAnswer\else\AMCmessage{MULT}\fi
1230 \begin{tikz-multi}[o]
1231 \node[#3] (var) at (0,0) {#2};}
1232 {\end{tikz-multi}
1233 \end{question}}
1234 \newenvironment{questionmultx}[1]{%

```



```

1235 \begingroup\def\multiSymbole{}\begin{questionmult}{#1}}%
1236 {\end{questionmult}}\endgroup}
1237 \newdimen\ouverte@vs
1238 \newenvironment{questionouverte}[1][3cm]{%
1239 \AMC@stepQuestion%
1240 \AMCtype@multifalse\ouverte@vs=#1%
1241 \ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1242 \AMCbeginQuestion{\AMC@qaff}{}}%
1243 {\vspace*{\ouverte@vs}\ifAMC@qbloc\end{minipage}\vspace{3ex}\fi}

```

#### 4.11.6 Explanations

`\explain` The command `\explain{<text>}` is used inside question-like environments to give the explanation for the answers of a question. The command `\explaincontext{<text>}` inserts its argument only in the corrected paper.

```

1244 \newcommand{\explain}[1]{%
1245 \ifAMC@correc%
1246 \AMCif@env{question}{\par\noindent{\AMC@loc@explain #1}}{\AMC@error@explain}\vspace{1ex}%
1247 \else%
1248 \AMCif@env{question}{}{\AMC@error@explain}%
1249 \fi%
1250 }
1251 \newcommand{\explaincontext}[1]{%
1252 \ifAMC@correc%
1253 #1%
1254 \fi%
1255 }

```

#### 4.12 Scoring

`\scoring` Scoring strategies are simply transmitted to the .amc file for later analysis.

`\scoringDefaultS` `\scoring{<score>}` details the scoring strategy for current question or current answer, `\scoringDefaultM` `\scoringDefaultS{<score>}` and `\scoringDefaultM{<score>}` gives default scoring strategy for simple and multiple questions, and `\QuestionIndicative` tells that the current question is not to be taken into account in the global mark.

```

1256 \def\scoring#1{\ifAMC@calibration\AMCmessage{B=#1}\fi\ignorespaces}
1257 \def\scoringDefaultS#1{\ifAMC@calibration\AMCmessage{BDS=#1}\fi}
1258 \def\scoringDefaultM#1{\ifAMC@calibration\AMCmessage{BDM=#1}\fi}
1259 \def\QuestionIndicative{\ifAMC@calibration\AMCmessage{INDIC}\fi}

```

## 4.13 Numerical data

### 4.13.1 Codes

`\AMCcodeGrid` Students can code some numerical information (such as student number) through special questions, which can be formatted easily with the command `\AMCcodeGrid[⟨opts⟩]{⟨key⟩}{⟨descr⟩}`, where `⟨key⟩` is a key prefix and `⟨descr⟩` is a coma-separated list of character pools to offer. The characters entered by the student will be available through the questions `⟨key⟩[1], …, ⟨key⟩[⟨length(descr)⟩]`.

As an example,

`\AMCcodeGrid{code}{ABCD,012345,012345,012345,012345}` produces the opposite boxes (two results are show here: without or with `separateanswersheet` option), and trace positions of all the boxes in the `.xy` file with the `code` identifier: the first digit is represented by question with key `code[6]`, the second by question with key `code[5]`, and so on.

Positions of the boxes are logged in the `.xy` file, as shown in section 5.3 for the first set of boxes (without `separateanswersheet`, with digits outside boxes).

	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>
	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>
<input type="text"/>	A	<input type="text"/>	2	<input type="text"/>	2	<input type="text"/>	2	<input type="text"/>	2
<input type="text"/>	B	<input type="text"/>	3	<input type="text"/>	3	<input type="text"/>	3	<input type="text"/>	3
<input type="text"/>	C	<input type="text"/>	4	<input type="text"/>	4	<input type="text"/>	4	<input type="text"/>	4
<input type="text"/>	D	<input type="text"/>	5	<input type="text"/>	5	<input type="text"/>	5	<input type="text"/>	5

  

	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>
	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>
<input type="text"/>	A	<input type="text"/>	2	<input type="text"/>	2	<input type="text"/>	2	<input type="text"/>	2
<input type="text"/>	B	<input type="text"/>	3	<input type="text"/>	3	<input type="text"/>	3	<input type="text"/>	3
<input type="text"/>	C	<input type="text"/>	4	<input type="text"/>	4	<input type="text"/>	4	<input type="text"/>	4
<input type="text"/>	D	<input type="text"/>	5	<input type="text"/>	5	<input type="text"/>	5	<input type="text"/>	5

The “horizontal” version can also be considered using option `h`, especially with a small number of digits. See opposite for the result of

`\AMCcodeGrid[h]{code}{ABCDEF,0123456789,0123456789}`.

The `\AMCcodeGridInt[⟨opts⟩]{⟨key⟩}{⟨n⟩}` is a shortcut for calling `\AMCcodeGrid` with `⟨n⟩` digits from 0 to 9. This allows to create grids for `⟨n⟩`-digits integers easily.

These two commands supports the following options (given as a comma-separated list optional argument `⟨opts⟩`):

- `vertical=true` or `false` to indicate the direction to be used (default is `true`);
- `h` is a shortcut for `vertical=false`;
- `v` is a shortcut for `vertical=true`;
- `top` to request top-aligned columns in vertical direction.
- `multi` for codes that are repeated on each page.

```

1260 \newcount\AMC@chiffres
1261 \newdimen\AMCcodeHspace\AMCcodeHspace=.5em
1262 \newdimen\AMCcodeVspace\AMCcodeVspace=.5em
1263 \newcommand\AMCcodeID@squarebrackets[2]{#1[#2]}
1264 \newcommand\AMCcodeID@dot[2]{#1.#2}
1265 \newcommand\AMCcodeID@@[1]{%
1266   \expandafter\def\expandafter\AMCcodeID\expandafter{\csname AMCcodeID@#1\endcsname}%
1267   \def\AMCcodeID@mode{#1}%
1268 }
```

```

1269 \AMCcodeID@@{squarebrackets}
1270 \ExplSyntaxOn
1271
1272 \clist_new:N \amc_code_descr_clist
1273 \seq_new:N \amc_code_digits_seq
1274 \int_new:N \amc_code_digit_n_int
1275 \bool_new:N \amc_code_vertical_bool
1276 \bool_new:N \amc_code_top_bool
1277 \bool_new:N \amc_code_multi_bool
1278 \clist_new:N \amc__multi_clist
1279
1280 \cs_new:Npn \amc_code_init:N #1 {
1281   \def\AMCbeginQuestion##1##2{}
1282   \def\AMCbeforeQuestion{}
1283   \AMCnoScoreZone
1284   \AMCquestionNumberfalse
1285   \setlength{\parindent}{0pt}
1286   \AMCnobloc
1287   \int_set:Nn \amc_code_digit_n_int { \clist_count:N #1 }
1288 }
1289
1290 \cs_new:Nn \amc_code_digit_init: {
1291   \QuestionIndicative
1292   \global\AMCrep@count=\z@
1293 }
1294
1295 \cs_new:Npn \amc_code_digit:n #1 {
1296   \global\advance\AMCrep@count\@ne\relax
1297   \ifAMC@calibration\AMCmessage{ REP = \the\AMCrep@count : M }\fi
1298   \hbox{\AMC@keyBox@{#1}{1}{case : \AMCid@name : \the\AMCid@quest , \the\AMCrep@count}}
1299   \bool_if:NTF \amc_code_vertical_bool {
1300     \vspace{\AMCcodeVspace}
1301   }{
1302     \hspace{\AMCcodeHspace}
1303   }
1304 }
1305
1306 \keys_define:nn { amccode } {
1307   vertical .bool_set:N = \amc_code_vertical_bool,
1308   vertical .initial:n = { true },
1309   vertical .default:n = { true },
1310   v .code:n = { \bool_set_true:N \amc_code_vertical_bool },
1311   h .code:n = { \bool_set_false:N \amc_code_vertical_bool },
1312   top .bool_set:N = \amc_code_top_bool,
1313   top .initial:n = { false },
1314   top .default:n = { true },
1315   multi .bool_set:N = \amc_code_multi_bool,
1316   multi .initial:n = { false },
1317   multi .default:n = { true }
1318 }
1319
1320 \cs_new_nopar:Nn \amc_multi_report: {
1321   \ifAMC@calibration

```

```

1322 \immediate\write\AMC@XYFILE{\string\with{multi=\clist_use:Nn\amc__multi_clist{,}}}}
1323 \fi
1324 }
1325 \cs_new_eq:NN \AMC@multi@report \amc_multi_report:
1326 \int_new:N \amc_multi_count_int
1327 \cs_new_nopar:Nn \amc_multi_clear: {
1328   \int_gzero:N \amc_multi_count_int
1329 }
1330 \cs_new_eq:NN \AMC@multiclear \amc_multi_clear:
1331
1332 \cs_new:Npn \amc_code_generate:nNn #1#2#3 {
1333   { \keys_set:nn { amccode } { #3 }
1334     \bool_if:NTF \amc_code_multi_bool {
1335       \clist_gset:Nn \amc__multi_clist { #1 }
1336     } {}
1337     \bool_if:NTF \amc_code_multi_bool { \int_gincr:N \amc_multi_count_int } {}
1338     \amc_code_init:N #2
1339     \clist_map_inline:Nn #2 { % iterates over 'digits'
1340       \begin{question}{
1341         \AMCcodeID{ #1 \bool_if:NTF
1342           \amc_code_multi_bool
1343           { * \int_use:N \amc_multi_count_int } {} }
1344         { \int_use:N \amc_code_digit_n_int }
1345       }
1346       \amc_code_digit_init:
1347       \seq_set_split:Nnn \amc_code_digits_seq {} { ##1 }
1348       \bool_if:NTF \amc_code_vertical_bool {
1349         \hspace{0pt}
1350         \bool_if:NTF \amc_code_top_bool { \vtop } { \vbox }
1351         \bgroup
1352       }{
1353         \hbox\bgroup
1354       }
1355       \seq_map_inline:Nn \amc_code_digits_seq {
1356         % iterates over available characters for 'digit'
1357         \amc_code_digit:n { ####1 }
1358       }
1359       \bool_if:NTF \amc_code_vertical_bool {
1360         \vspace{-\AMCcodeVspace}\egroup
1361         \hspace{\AMCcodeHspace}
1362       }{
1363         \egroup\vspace{\AMCcodeVspace}
1364         \par
1365       }
1366       \end{question}
1367       \int_decr:N \amc_code_digit_n_int
1368     }
1369   }
1370 }
1371
1372 \cs_new:Npn \amc_code_generate:nnn #1#2#3 {
1373   \clist_set:Nn \amc_code_descr_clist { #2 }
1374   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }

```

```

1375 }
1376 \cs_generate_variant:Nn \amc_code_generate:nnn { xxx }
1377 \newcommand{\AMCcodeGrid}[3][]{
1378   \amc_code_generate:xxx { #2 } { #3 } { #1 }
1379 }
1380
1381 \cs_new:Npn \amc_code_generate_integer:nnn #1#2#3 {
1382   \clist_clear:N \amc_code_descr_clist
1383   \prg_replicate:nn { #2 } { \clist_put_right:Nn \amc_code_descr_clist { 0123456789 } }
1384   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1385 }
1386 \cs_generate_variant:Nn \amc_code_generate_integer:nnn { xxx }
1387 \newcommand{\AMCcodeGridInt}[3][]{
1388   \amc_code_generate_integer:xxx { #2 } { #3 } { #1 }
1389 }
1390
1391 \cs_new:Npn \amc_code_generate_integer_v:nn #1#2 {
1392   \amc_code_generate_integer:nnn { #1 } { #2 } { v }
1393 }
1394 \cs_new:Npn \amc_code_generate_integer_h:nn #1#2 {
1395   \amc_code_generate_integer:nnn { #1 } { #2 } { h }
1396 }
1397 \cs_generate_variant:Nn \amc_code_generate_integer_v:nn { xx }
1398 \cs_generate_variant:Nn \amc_code_generate_integer_h:nn { xx }
1399 \cs_new_eq:NN \AMCcode \amc_code_generate_integer_v:xx
1400 \cs_new_eq:NN \AMCcodeH \amc_code_generate_integer_h:xx
1401
1402 \ExplSyntaxOff

```

#### 4.13.2 Numerical questions

`\AMCnumericChoices` The command `\AMCnumericChoices{<correct>}{<options>}` can be used as a replacement for the `choices` environment when the questions asks for a numeric value to code on the answer sheet.

As an example,

```

\begin{question}{product}
  What is the value of  $7 \times 5$ ?
  \AMCnumericChoices{35}{digits=2,sign=false}
\end{question}

```

produces (in correction mode):

<b>Question 11</b> What is the value of $7 \times 5$ ?
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> <input type="checkbox"/>0 <input type="checkbox"/>1 <input type="checkbox"/>2 <input checked="" type="checkbox"/>3 <input type="checkbox"/>4 <input type="checkbox"/>5 <input type="checkbox"/>6 <input type="checkbox"/>7 <input type="checkbox"/>8 <input type="checkbox"/>9  <input type="checkbox"/>0 <input type="checkbox"/>1 <input type="checkbox"/>2 <input type="checkbox"/>3 <input type="checkbox"/>4 <input checked="" type="checkbox"/>5 <input type="checkbox"/>6 <input type="checkbox"/>7 <input type="checkbox"/>8 <input type="checkbox"/>9         </div>

and these boxes are only shown on the separate answer sheet if the `separateanswersheet` option is used.

This command uses the `\AMCformatChoices{<showcommand>}{<hidecommand>}{<arg1>}{<arg2>}` command, that calls either `<hidecommand>{<arg1>}{<arg2>}` if the `separateanswersheet` op-

tion is used and if we are currently in the question part (not in the answer sheet), or  $\langle showcommand \rangle \{ \langle arg1 \rangle \} \{ \langle arg2 \rangle \}$  when all the boxes are to be produced.

```

1403 \newcommand\AMCformatChoices[4]{%
1404   \global\AMCrep@count=\z@%
1405   \AMC@if@separate@question{%
1406     \AMC@mem@add{\global\AMCrep@count=\z@%
1407       #1{#3}{#4}}%
1408   }%
1409   \ifAMC@ensemble%
1410     #2{#3}{#4}%
1411     \AMCmessage{QPART}%
1412   \else%
1413     #1{#3}{#4}%
1414   \fi%
1415 }
```

Some computation commands are now defined. The command  $\backslash amc\_fp\_decompose:NNn\{\langle fp\ var \rangle\}\{\langle int\ var \rangle\}\{\langle x \rangle\}$  sets  $\langle fp\ var \rangle$  to be the *mantissa* and  $\langle int\ var \rangle$  the *exponent* of the floating point number  $\langle x \rangle$ . For example,  $\backslash amc\_fp\_decompose:NNn\mant\_fp\expo\_int\{123.456\}$  give the value 1.23456 to  $\backslash mant\_fp$  and 2 to  $\backslash expo\_int$  (because  $123.456 = 1.23456 \times 10^2$ ).

The command  $\backslash amc\_fp\_to\_digits:Nnnn\{\langle clist \rangle\}\{\langle x \rangle\}\{\langle n\ digits \rangle\}\{\langle base \rangle\}$  rounds the floating point number  $\langle x \rangle$  and populates the comma separated list  $\langle clist \rangle$  with its  $\langle n\ digits \rangle$  digits in base  $\langle base \rangle$ . An error is issued if  $\langle x \rangle$  would have required more digits.

```

1416 \ExplSyntaxOn
1417
1418 \cs_generate_variant:Nn \tl_replace_once:Nnn { Nxn }
1419
1420 \tl_new:N \amc_ee_tl
1421 \seq_new:N \amc_ee_seq
```

Note that with some versions of 13fp-convert (prior to 2017-09-18),  $\backslash fp\_to\_scientific$  leads to a ‘e’ with catcode 12 (*other*). We convert it to catcode *letter* before splitting.

```

1422 \group_begin:
1423 \char_set_catcode_other:N E
1424 \tex_lowercase:D
1425 {
1426   \cs_new:Npn \amc_read_scientific:NNn #1 #2 #3 {
1427     \tl_set:Nn \amc_ee_tl { #3 }
1428     \tl_replace_once:Nxn \amc_ee_tl { E } { e }
1429     \seq_set_split:NnV \amc_ee_seq e \amc_ee_tl
1430     \fp_set:Nn #1 { \seq_item:Nn \amc_ee_seq 1 }
1431     \int_set:Nn #2 { \seq_item:Nn \amc_ee_seq 2 }
1432   }
1433 }
1434 \group_end:
1435
1436 \cs_generate_variant:Nn \amc_read_scientific:NNn { NNf, NNx }
1437
1438 \fp_new:N \amc_fulls_fp
1439 \cs_new:Npn \amc_fp_decompose:NNn #1 #2 #3 {
1440   \fp_set:Nn \amc_fulls_fp { #3 }
```

Note that with some versions of 13fp-convert, the exponent part is omitted for some values, so that we add e 0.

```

1441 \amc_read_scientific:NNx #1 #2
1442 { \fp_to_scientific:N \amc_fulls_fp e 0 }
1443 }
1444 \cs_generate_variant:Nn \amc_fp_decompose:NNn { NNx }
1445
1446 \fp_new:N \amc_num_mantissa_fp
1447 \int_new:N \amc_num_exponent_int
1448 \cs_new:Npn \amc_fp_n_significant_digits:Nnn #1 #2 #3 {
1449   \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_exponent_int
1450   { #2 }
1451   \fp_set:Nn #1
1452   { round(\amc_num_mantissa_fp * 10^((#3)-1)) }
1453   \fp_compare:nTF { abs(#1) >= 10^(#3) }
1454   {
1455     \fp_set:Nn #1 { #1 / 10 }
1456   } { }
1457 }
1458
1459 \fp_new:N \amc_num_nsig_fp
1460 \cs_new:Npn \amc_fp_show_n_significant_digits:nn #1 #2 {
1461   \amc_fp_n_significant_digits:Nnn \amc_num_nsig_fp { #1 } { #2 }
1462 }
1463 \cs_new_eq:NN \AMCsignificantDigits \amc_fp_show_n_significant_digits:nn
1464
1465 \cs_new:Npn \amc_fp_show_significant_digits: {
1466   \fp_use:N \amc_num_nsig_fp
1467 }
1468 \cs_new_eq:NN \AMCshowSignificantDigits \amc_fp_show_significant_digits:
1469
1470 \cs_new:Npn \amc_fp_n_digits:Nnn #1 #2 #3 {
1471   \fp_set:Nn #1
1472   { round((#2) * \amc_num_base_int^(#3)) }
1473 }
1474
1475 \int_new:N \amc_todigits_int
1476 \cs_new:Npn \amc_fp_to_digits:Nnnn #1 #2 #3 #4 {
1477   \clist_clear:N #1
1478   \int_set:Nn \amc_todigits_int { \fp_eval:n { abs(round(#2)) } }
1479   \prg_replicate:nn { #3 } {
1480     \clist_put_left:Nx #1 { \int_mod:nn \amc_todigits_int { #4 } }
1481     \int_set:Nn \amc_todigits_int
1482     { \int_div_truncate:nn \amc_todigits_int { #4 } }
1483   }
1484   \int_compare:nNnTF \amc_todigits_int = 0 { } {
1485     \message{^^J!~Error:~number~too~large,
1486       ~some~digits~will~be~discarded^^J}
1487   }
1488 }
1489
1490 \cs_new:Npn \amc_invalid_digits:Nn #1 #2 {
1491   \clist_clear:N #1
1492   \prg_replicate:nn { #2 } { \clist_put_left:Nx #1 { -1 } }
1493 }

```

```

1494
1495 \cs_new:Npn \amc_get_fp_sign:Nn #1 #2 {
1496   \fp_compare:nNnTF #2 < 0 {
1497     \int_set:Nn #1 { -1 }
1498   }{
1499     \fp_compare:nNnTF #2 > 0 {
1500       \int_set:Nn #1 { 1 }
1501     }{
1502       \int_set:Nn #1 { 0 }
1503     }
1504   }
1505 }
1506
1507 \cs_new:Npn \amc_get_int_sign:Nn #1 #2 {
1508   \int_compare:nNnTF #2 < 0 {
1509     \int_set:Nn #1 { -1 }
1510   }{
1511     \int_compare:nNnTF #2 > 0 {
1512       \int_set:Nn #1 { 1 }
1513     }{
1514       \int_set:Nn #1 { 0 }
1515     }
1516   }
1517 }
1518
1519 \ExplSyntaxOff

```

The command `\AMCnumericShow{⟨value⟩}{⟨opts⟩}` is called to draw all necessary boxes to code a numerical value `⟨value⟩` with options given as a comma separated list `⟨opts⟩`. `\AMCnumericOpts{⟨opts⟩}` can be used to set some default values for these options.

Begin with the available options:

```

1520 \def\AMCnntextGoto{}
1521 \def\AMCnntextVHead#1{\emph{b#1}}
1522 \newdimen\AMCnumeric@Hspace\AMCnumeric@Hspace=.5em
1523 \newdimen\AMCnumeric@Vspace\AMCnumeric@Vspace=1ex
1524 \ExplSyntaxOn
1525
1526 \keys_define:nn { amcnumeric } {
1527   Tsign .code:n = {\def\AMCnntextSign{#1}},
1528   Tsign .initial:n = {},
1529   Tpoint .code:n = {\def\AMCdecimalPoint{#1}},
1530   Tpoint .initial:n = { \raisebox{1ex}{\bf .} },
1531   Texponent .code:n = {\def\AMCexponent{#1}},
1532   Texponent .initial:n = { $\times 10^{\textasciicircum} $ },
1533   vspace .code:n = {\AMCnumeric@Vspace=#1},
1534   hspace .code:n = {\AMCnumeric@Hspace=#1},
1535   bordercol .code:n = {\def\AMCncol@Border{#1}},
1536   bordercol .initial:n = { lightgray },
1537   borderwidth .code:n = {\def\AMCncol@BorderWidth{#1}},
1538   borderwidth .initial:n = { 1mm },
1539   backgroundcol .code:n = {\def\AMCncol@Background{#1}},
1540   backgroundcol .initial:n = { white },
1541   digits .int_set:N = \amc_num_ndigits_int,
1542   digits .initial:n = { 3 },

```



```

1543 decimals .int_set:N = \amc_num_decd_int,
1544 decimals .initial:n = { 0 },
1545 exponent .int_set:N = \amc_num_expo_int,
1546 exponent .initial:n = { 0 },
1547 base .int_set:N = \amc_num_base_int,
1548 base .initial:n = { 10 },
1549 sign .bool_set:N = \amc_num_sign_bool,
1550 sign .initial:n = { true },
1551 sign .default:n = { true },
1552 exposign .bool_set:N = \amc_num_exposign_bool,
1553 exposign .initial:n = { true },
1554 exposign .default:n = { true },
1555 strict .bool_set:N = \amc_num_strict_bool,
1556 strict .initial:n = { false },
1557 strict .default:n = { true },
1558 scoring .bool_set:N = \amc_num_scoring_bool,
1559 scoring .initial:n = { true },
1560 scoring .default:n = { true },
1561 ignoreblank .bool_set:N = \amc_num_ignoreblank_bool,
1562 ignoreblank .initial:n = { false },
1563 ignoreblank .default:n = { true },
1564 vertical .bool_set:N = \amc_num_vertical_bool,
1565 vertical .initial:n = { false },
1566 vertical .default:n = { true },
1567 expovertical .bool_set:N = \amc_num_expovertical_bool,
1568 expovertical .initial:n = { false },
1569 expovertical .default:n = { true },
1570 reverse .bool_set:N = \amc_num_reverse_bool,
1571 reverse .initial:n = { false },
1572 reverse .default:n = { true },
1573 vhead .bool_set:N = \amc_num_vhead_bool,
1574 vhead .initial:n = { false },
1575 vhead .default:n = { true },
1576 Tvhead .code:n = {\clist_set:Nx \amc_tvhead_clist {\clist_reverse:n {#1}}},
1577 Tvhead .initial:n = {}, % \c_empty_clist does not work with \clist_reverse:n
1578 vheadunitindex .int_set:N = \amc_vheadunitindex_int,
1579 vheadunitindex .initial:n = 0,
1580 nozero .bool_set:N = \amc_num_nozero_bool,
1581 nozero .initial:n = { false },
1582 nozero .default:n = { true },
1583 significant .bool_set:N = \amc_num_significant_bool,
1584 significant .initial:n = { false },
1585 significant .default:n = { true },
1586 scoreexact .code:n = {\def\AMC@numeric@scoreexact{#1}},
1587 scoreexact .initial:n = { 2 },
1588 scoreapprox .code:n = {\def\AMC@numeric@scoreapprox{#1}},
1589 scoreapprox .initial:n = { 1 },
1590 scorewrong .code:n = {\def\AMC@numeric@scorewrong{#1}},
1591 scorewrong .initial:n = { 0 },
1592 exact .int_set:N = \amc_num_exact_int,
1593 exact .initial:n = { 0 },
1594 approx .int_set:N = \amc_num_approx_int,
1595 approx .initial:n = { 0 },

```

```

1596 keepas .code:n = {\def\AMC@numeric@keepas{#1}},
1597 keepas .initial:n = {},
1598 alsocorrect .code:n = {\def\AMC@numeric@alsocorrect{#1}},
1599 alsocorrect .initial:n = {}
1600 }
1601
1602 \cs_new:Npn \amc_num_setopt #1 {
1603   \keys_set:nn { amcnumeric } { #1 }
1604 }
1605
1606 \cs_new_nopar:Nn \amc_num_check_score_opts: {
1607   \bool_if:NTF \amc_num_ignoreblank_bool {
1608     \int_compare:nNnTF \amc_num_base_int = { 10 } { } {
1609       \message{^^J!~Error:~ignoreblank~can~only~be~used~with~number~base~10^^J}
1610     }
1611   } {}
1612 }
1613
1614 \cs_new_eq:NN \AMCnumericOpts \amc_num_setopt
1615

```

The command `\amc_num_char:nn{<i>}<j>}` draw a box with content *<i>* (only if needed), where *<j>* is `\AMC@checkbox` if the corresponding choice is correct and empty if not.

```

1616 \cs_new:Npn \amc_num_char:nn #1 #2 {
1617   \global\advance\AMCrep@count\@ne\relax
1618   \AMCmessage{REP= \the\AMCrep@count :
1619     \ifx#2\AMC@checkbox B\else M\fi }
1620   \ifAMC@correc
1621     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :
1622       \the\AMCid@quest , \the\AMCrep@count}
1623   \else
1624     \protect\AMC@keyBox@{#1}{}{1}{case : \AMCid@name :
1625       \the\AMCid@quest , \the\AMCrep@count}
1626   \fi
1627 }

```

The command `\amc_num_digit_box:nn{<i>}<j>}` draws a box for current digit value *<i>*, where *<j>* is the correct value for the current digit. If *<i>* is greater than 9, it is converted to a character from the English alphabet (A for 10, B for 11...)

```

1628 \int_new:N \amc_num_digit_value_int
1629 \tl_new:N \amc_num_digit_value_tl
1630 \cs_new:Npn \amc_num_digit_box:nn #1 #2 {
1631   \int_set:Nn \amc_num_digit_value_int { #1 }
1632   \tl_set:Nn \amc_num_digit_value_tl {
1633     \int_compare:nNnTF { \amc_num_digit_value_int } < { 10 }
1634     { \int_to_arabic:n { \amc_num_digit_value_int } }
1635     { \int_to_Alph:n { \amc_num_digit_value_int - 9 } }
1636   }
1637   \int_compare:nNnTF { #1 } = { #2 } {
1638     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1639     {\AMC@checkbox}
1640   } {
1641     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }

```

```

1642         {}
1643     }
1644 }

```

The command `\amc_num_sign_boxes:Nn{<sign>}{<prefix>}` draws two boxes for the students to code the sign (with a right value given by the boolean `<negative>`).

```

1645 \cs_new:Npn \amc_num_sign_boxes:Nn #1 #2 {
1646   \int_case:nn { #1 } {
1647     { -1 } {
1648       \hbox{\amc_num_char:nn{+}{}}
1649       \vspace{\AMCnumeric@Vspace}
1650       \AMCmessage{B=set. sign #2 =1}
1651       \hbox{\amc_num_char:nn{-}{\AMC@checkbox}}
1652       \AMCmessage{B=set. sign #2 =-1}
1653     }
1654     { 1 } {
1655       \hbox{\amc_num_char:nn{+}{\AMC@checkbox}}
1656       \vspace{\AMCnumeric@Vspace}
1657       \AMCmessage{B=set. sign #2 =1}
1658       \hbox{\amc_num_char:nn{-}{}}
1659       \AMCmessage{B=set. sign #2 =-1}
1660     }
1661     { 0 } {
1662       \hbox{\amc_num_char:nn{+}{}}
1663       \vspace{\AMCnumeric@Vspace}
1664       \AMCmessage{B=set. sign #2 =1}
1665       \hbox{\amc_num_char:nn{-}{}}
1666       \AMCmessage{B=set. sign #2 =-1}
1667     }
1668   }
1669 }

```

The command `\amc_num_digit_boxes_h:nnn{<varname>}{<correct>}{<maxdigit>}` draws a serie of boxes for all possible values of a digit (from 0 to `<maxdigit>`), where the correct value is `<correct>`, transmitting scoring data to AMC so that the variable `<varname>` will be set to the value chosen by the student.

```

1670 \cs_new:Npn \amc_num_digit_boxes_h:nnn #1 #2 #3 {
1671   \int_step_inline:nnnn
1672   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1673   { 1 } { #3 - 1 } {
1674     \amc_num_digit_box:nn { ##1 }{ #2 }
1675     \AMCmessage{B= set. #1 = ##1}
1676     \hspace{\AMCnumeric@Hspace}
1677   }
1678   \hspace{-\AMCnumeric@Hspace}
1679 }
1680
1681 \cs_new:Npn \amc_num_digit_boxes_v:nnn #1 #2 #3 {
1682   \int_step_inline:nnnn
1683   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1684   { 1 } { #3 - 1 } {
1685     \vbox{\hbox{
1686       \amc_num_digit_box:nn { ##1 }{ #2 }
1687     }}

```

```

1688 \AMCmessage{B= set. #1 = ##1}
1689 \int_compare:nNnTF { ##1 } < { #3 - 1 } {
1690   \vspace{\AMCnumeric@Vspace}
1691 } {}
1692 }
1693 }
1694
1695 \int_new:N \amc_num_first_digit_int
1696 \cs_new:Npn \amc_num_digit_boxes_vr:nnn #1 #2 #3 {
1697   \int_set:Nn \amc_num_first_digit_int
1698   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1699   \int_step_inline:nnnn { #3 - 1 } { -1 }
1700   \amc_num_first_digit_int {
1701     \vbox{\hbox{
1702       \amc_num_digit_box:nn { ##1 } { #2 }
1703     }}
1704     \AMCmessage{B= set. #1 = ##1}
1705     \int_compare:nNnTF { ##1 } > \amc_num_first_digit_int {
1706       \vspace{\AMCnumeric@Vspace}
1707     } {}
1708   }
1709 }

```

The command `\amc_num_integer_boxes_v:Nnn{<correct digits>}{<prefix>}{<decimals>}` draws boxes for integer entry, without the sign.

```

1710 \cs_new:Npn \amc_num_integer_boxes_v:Nnn #1 #2 #3 {
begin a loop over all digits,
1711   \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1712   \clist_map_inline:Nn #1 {
place the decimal point if necessary,
1713     \int_compare:nNnTF \amc_num_digit_int = { #3 } {
1714       \hbox{ \AMCdecimalPoint } \hspace{\AMCnumeric@Hspace}
1715     } {}
draw the box for this digit,
1716     \hbox{\vbox{
1717       \bool_if:NTF \amc_num_vhead_bool {
1718         \vbox{\hbox to \AMC@boxedwidth{\hfill\AMCncontextVHead{ \int_eval:n
1719           { \amc_num_digit_int - 1 } }\hfill}}
1720         \vspace{\AMCnumeric@Vspace}
1721       } {}
1722       \bool_if:NTF \amc_num_reverse_bool {
1723         \amc_num_digit_boxes_vr:nnn { #2
1724           \int_to_Alph:n \amc_num_digit_int }
1725         { ##1 } { \amc_num_base_int }
1726       } {
1727         \amc_num_digit_boxes_v:nnn { #2
1728           \int_to_Alph:n \amc_num_digit_int }
1729         { ##1 } { \amc_num_base_int }
1730       }
1731     }}

```

and end the loop over digits, adding space if this is not the last one.

```

1732 \int_compare:nNnTF \amc_num_digit_int > 1 {
1733   \hspace{\AMCnumeric@Hspace}
1734 } { }
1735 \int_decr:N \amc_num_digit_int
1736 }
1737 }
1738

```

The command `\amc_num_integer_boxes_h:Nnn{<correct digits>}{<prefix>}{<decimals>}` does the same, in horizontal mode.

```

1739
1740 \cs_new:Npn \amc_num_integer_boxes_h:Nnn #1 #2 #3 {
1741   \vbox{
1742     \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1743     \clist_map_inline:Nn #1 {
1744       \int_compare:nNnTF
1745         \amc_num_digit_int = { #3 } {
1746         \hbox{ \AMCdecimalPoint }
1747       } { }
1748       \hbox{
1749         \amc_num_digit_boxes_h:nnn { #2
1750           \int_to_Alph:n \amc_num_digit_int }
1751         { ##1 } \amc_num_base_int
1752       }
1753       \int_compare:nNnTF \amc_num_digit_int > 1 {
1754         \vspace{\AMCnumeric@Vspace}
1755       } { }
1756       \int_decr:N \amc_num_digit_int
1757     }
1758   }
1759

```

Finally, `\amc_num_integer_boxes:NnnNN{<correct digits>}{<prefix>}{<decimals>}{<sign bool>}{<sign>}` draws boxes for integer entry, including the sign if `<sign bool>` is true. When using the `strict` option, check the `+` box for a null value.

```

1760
1761 \cs_new:Npn \amc_num_integer_boxes:NnnNN #1 #2 #3 #4 #5 {
1762   \hbox{
1763     \bool_if:NTF { #4 } {
1764       \vbox{
1765         \ifx\AMCnTextSign\@empty\@empty\else
1766         \hbox{\AMCnTextSign}\vspace{\AMCnumeric@Vspace}\fi
1767         \bool_if:NTF \amc_num_strict_bool {
1768           \int_compare:nNnTF { #5 } = 0 {
1769             \amc_num_sign_boxes:Nn { 1 } { #2 }
1770           }{
1771             \amc_num_sign_boxes:Nn { #5 } { #2 }
1772           }
1773         }{
1774           \amc_num_sign_boxes:Nn { #5 } { #2 }
1775         }
1776       }
1777       \hspace{.5em}
1778       \vrule

```

```

1779     \hspace{.5em}
1780   } { }
1781   \hbox{
1782     \bool_if:NTF \amc_num_vertical_bool
1783     \amc_num_integer_boxes_v:Nnn \amc_num_integer_boxes_h:Nnn
1784     #1 { #2 } { #3 }
1785   }
1786 }
1787 }
1788

```

The command `\amc_num_build_integer_scoring:Nnnnn{<tl var>}{<sign bool>}{<prefix>}{<n>}{<decimals>}` builds a scoring to compute an integer from a serie of  $\langle n \rangle$ -digits boxes (from which  $\langle decimals \rangle$  are for decimals), with name prefix  $\langle prefix \rangle$ , using a sign variable if  $\langle sign bool \rangle$  is true.

```

1789
1790 \cs_new:Npn \amc_num_build_integer_scoring:Nnnnn #1 #2 #3 #4 #5 {
1791   \amc_num_check_score_opts:
1792   \tl_clear:N #1
1793   \int_set_eq:NN \amc_num_digit_int { #4 }
1794   \int_while_do:nNnn \amc_num_digit_int > 0 {
1795     \bool_if:NTF \amc_num_strict_bool {
1796       \AMCmessage{B=requires. #3
1797         \int_to_Alph:n \amc_num_digit_int = 1}
1798     } {
1799       \AMCmessage{B=default. #3
1800         \int_to_Alph:n \amc_num_digit_int =
1801         \bool_if:NTF \amc_num_ignoreblank_bool { } { 0 }
1802       }
1803     }
1804     \int_compare:nNnTF \amc_num_digit_int = #4 { } {
1805       \bool_if:NTF \amc_num_ignoreblank_bool {
1806         \tl_put_right:Nx #1 { ~.~ }
1807         \int_compare:nNnTF \amc_num_digit_int = #5 {
1808           \tl_put_right:Nx #1 { "." ~.~ }
1809         } { }
1810       } {
1811         \tl_put_left:Nn #1 { ( }
1812         \tl_put_right:Nx #1 { ) *
1813         \int_use:N \amc_num_base_int + }
1814       }
1815     }
1816     \tl_put_right:Nx #1
1817     { #3 \int_to_Alph:n \amc_num_digit_int }
1818     \int_decr:N \amc_num_digit_int
1819   }
1820   \bool_if:NTF \amc_num_ignoreblank_bool {
1821     \tl_put_left:Nn #1 { ( 0 + ( }
1822     \tl_put_right:Nn #1 { ) ) }
1823     \int_compare:nNnTF \amc_num_decd_int > 0 {
1824       \tl_put_right:Nx #1 { * ( \int_use:N \amc_num_base_int ** \int_eval:n { #5 } ) }
1825     } { }
1826   } {
1827     \tl_put_left:Nn #1 { ( }
1828     \tl_put_right:Nn #1 { ) }

```

```

1829 }
1830 \bool_if:NTF { #2 } {
1831   \bool_if:NTF \amc_num_strict_bool {
1832     \AMCmessage{B=requires. sign #3 =1}
1833   } {
1834     \AMCmessage{B=default. sign #3 =1}
1835   }
1836   \tl_put_right:Nx #1 { * ( sign #3 ) }
1837 } { }
1838 }
1839

```

Then the command `\AMCnumericShow{<x>}{<options>}` itself:

```

1840
1841 \fp_new:N \amc_num_result_fp
1842 \fp_new:N \amc_num_correct_fp
1843 \clist_new:N \amc_num_digits_clist
1844 \clist_new:N \amc_num_expo_digits_clist
1845 \int_new:N \amc_num_digit_int
1846 \int_new:N \amc_num_sign_int
1847 \int_new:N \amc_num_expo_sign_int
1848 \tl_new:N \amc_num_compute_tl
1849 \tl_new:N \amc_num_expo_tl
1850 \int_new:N \amc_num_correct_expo_int
1851
1852 \cs_new:Npn \amc_numeric_show:nn #1 #2 {

```

We have to tell AMC that the scoring we will give concerns this question:

```

1853 \ifAMC@ensemble\ifAMCformulaire@dedans
1854   \AMCmessage{Q=\the\AMCid@quest}
1855 \fi\fi

```

Then we parse the options from `<opts>`:

```

1856 {\keys_set:nn { amcnumeric } { #2 }

```

if Tvhead is set, we adapt the `AMCncontextVHead` macro

```

1857 \int_compare:nTF {\amc_vheadunitindex_int = 0}
1858 {\int_set:Nn \amc_vheadunitindex_int {\amc_num_decd_int + 1}}
1859 {}
1860
1861 \clist_if_empty:NTF \amc_tvhead_clist
1862 {}
1863 {
1864   \def\AMCncontextVHead##1{
1865     \int_set:Nn \l_tmpa_int
1866     {\int_max:nn
1867     {(##1) - \amc_num_decd_int + \amc_vheadunitindex_int}
1868     {0}}
1869     \emph{\clist_item:Nn \amc_tvhead_clist {\l_tmpa_int}}}}
1870
1871   \bool_if:nTF { \bool_if_p:N\amc_num_significant_bool
1872     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1873     \message{^^J!~AMCnumeric~Error:~significant=true~can't~be~used~with~base!=10.^^J}
1874   } { }
1875   \bool_if:nTF { \int_compare_p:n { \amc_num_expo_int != 0 }

```

```

1876     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1877     \message{^^J!~AMCnumeric~Error:~scientific~notation~can't~be~used~with~base!=10.^^J}
1878 } {}

```

Convert the floating point correct value to integer, taking into account the parameters significant, exponent and decimals:

```

1879 \ifx\@empty#1\@empty
1880   \fp_set:Nn \amc_num_correct_fp { 0 }
1881   \fp_set:Nn \amc_num_mantissa_fp { 0 }
1882   \int_set:Nn \amc_num_correct_expo_int { 0 }
1883 \else
1884 \bool_if:NTF \amc_num_significant_bool {
1885   \amc_fp_n_significant_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_ndigits_int
1886 } {
1887   \int_compare:nNnTF \amc_num_expo_int > 0 {
1888     \amc_fp_decompose:NnN \amc_num_mantissa_fp \amc_num_correct_expo_int { #1 }
1889     \int_compare:nNnTF { \amc_num_ndigits_int - \amc_num_decd_int } > 1 {
1890       \fp_set:Nn \amc_num_mantissa_fp {
1891         \amc_num_mantissa_fp * \amc_num_base_int^( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1892       }
1893       \int_set:Nn \amc_num_correct_expo_int {
1894         \amc_num_correct_expo_int - ( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1895       }
1896     } {}
1897     \amc_fp_n_digits:Nnn \amc_num_correct_fp \amc_num_mantissa_fp \amc_num_decd_int
1898   } {
1899     \amc_fp_n_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_decd_int
1900   }
1901 }
1902 \fi

```

Now extracts the required digits:

```

1903 \ifx\@empty#1\@empty
1904   \amc_invalid_digits:Nn \amc_num_digits_clist \amc_num_ndigits_int
1905   \amc_invalid_digits:Nn \amc_num_expo_digits_clist \amc_num_expo_int
1906   \int_set:Nn \amc_num_sign_int { 0 }
1907   \int_set:Nn \amc_num_expo_sign_int { 0 }
1908 \else
1909   \amc_fp_to_digits:Nnnn \amc_num_digits_clist \amc_num_correct_fp
1910   \amc_num_ndigits_int \amc_num_base_int
1911   \amc_get_fp_sign:Nn \amc_num_sign_int \amc_num_correct_fp
1912   \int_compare:nNnTF \amc_num_expo_int > 0 {
1913     \amc_fp_to_digits:Nnnn \amc_num_expo_digits_clist \amc_num_correct_expo_int
1914     \amc_num_expo_int \amc_num_base_int
1915     \amc_get_int_sign:Nn \amc_num_expo_sign_int \amc_num_correct_expo_int
1916   } {}
1917 \fi

```

The question scoring is given to AMC (if requested by the `scoring=true` option). Note that the variable `intV` refers to the correct value, and `intX` to the value entered by the student.

```

1918 \fp_set:Nn \amc_num_result_fp { #1 }
1919 \AMCmessage{B=numval=\fp_to_scientific:N \amc_num_result_fp ,
1920   numex=\int_use:N \amc_num_exact_int,
1921   numapp=\int_use:N \amc_num_approx_int,
1922   numsex=\AMC@numeric@scoreexact,

```



```

1923     num sapp=\AMC@numeric@scoreapprox
1924 }
1925 \bool_if:NTF \amc_num_scoring_bool {
1926   \AMCmessage{B=haut=mz=d=undef,p=undef,
1927     formula=(Vdifference <= \int_use:N \amc_num_exact_int ?
1928     \AMC@numeric@scoreexact :
1929     \int_compare:nNnTF \amc_num_approx_int = 0 {
1930       \AMC@numeric@scorewrong
1931     } {
1932       (Vdifference <= \int_use:N \amc_num_approx_int ?
1933       \AMC@numeric@scoreapprox : \AMC@numeric@scorewrong)
1934     }
1935   )}
1936 } {}
1937 \amc_num_build_integer_scoring:Nnnnn
1938   \amc_num_compute_tl \amc_num_sign_bool { digit } \amc_num_ndigits_int
1939   \amc_num_decd_int
1940   \int_compare:nNnTF \amc_num_expo_int > 0 {
1941     \amc_num_build_integer_scoring:Nnnnn
1942     \amc_num_expo_tl \amc_num_exposign_bool { expo } \amc_num_expo_int { 0 }
1943     \AMCmessage{B= set. intE = \amc_num_expo_tl}
1944   } {}
1945   \AMCmessage{B= set.intV = \fp_to_int:N \amc_num_correct_fp ,
1946     set.intXX = \amc_num_compute_tl }
1947   \int_compare:nNnTF \amc_num_expo_int > 0 {
1948     \AMCmessage{B= set.intX = intXX * \int_use:N \amc_num_base_int ** ( intE - (\int_use:N \amc_num_correc
1949   ){
1950     \AMCmessage{B= set.intX = intXX}
1951   }
1952   \int_compare:nNnTF \amc_num_expo_int > 0 {
1953     \AMCmessage{B= set.valueX = intXX * \int_use:N \amc_num_base_int ** (intE - \int_use:N \amc_num_decd
1954   ){
1955     \AMCmessage{B= set.valueX = intXX * \int_use:N \amc_num_base_int ** (- \int_use:N \amc_num_decd_int))}
1956   }
1957   \ifx\@empty\AMC@numeric@keepas\@empty\else
1958     \AMCmessage{B= setglobal.\AMC@numeric@keepas = valueX}
1959   \fi
1960   \ifx\@empty#1\@empty
1961     \bool_if:NTF \amc_num_significant_bool {
1962       \AMCmessage{B=set.Vdifference=0}
1963     }{
1964       \ifx\@empty\AMC@numeric@alsocorrect\@empty
1965         \AMCmessage{B=set.Vdifference=0}
1966       \else
1967         \AMCmessage{B="set.Vdifference =
1968           amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N \amc_num_decd_int, \int_use:N \amc_r
1969           "}
1970       \fi
1971     }
1972   \else
1973     \bool_if:NTF \amc_num_significant_bool {
1974       \AMCmessage{B=set.Vdifference="min( abs((intV)-(intX)) ,
1975         abs(\int_use:N \amc_num_base_int * (intV) - (intX)) ,

```

```

1976      abs((intV) - \int_use:N\amc_num_base_int * (intX)) )"}
1977  } {
1978    \ifx\@empty\AMC@numeric@alsocorrect\@empty
1979    \AMCmessage{B=set.Vdifference=abs((intV)-(intX))}
1980    \else
1981    \AMCmessage{B="set.Vdifference =
1982      min( amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\am
1983      abs((intV)-(intX)) )"}
1984    \fi
1985  }
1986  \fi

```

Begin now with the frame around all the boxes:

```

1987  \ifAMC@extractOnly\else
1988  \vspace{1.5ex}\par{
1989    \fboxrule=\AMCncol@BorderWidth
1990    \fcolorbox{\AMCncol@Border}{\AMCncol@Background}{
1991      \bool_if:NTF \amc_num_expovertical_bool {
1992        \hbox{\vbox{
1993          \vbox{\amc_num_integer_boxes:NnnNN
1994            \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1995            \amc_num_sign_int}
1996          \int_compare:nNnTF \amc_num_expo_int > 0 {
1997            \vspace{\AMCnumeric@Vspace}
1998            \vbox{\hbox{\AMCexponent}}
1999            \vspace{\AMCnumeric@Vspace}
2000            \vbox{\amc_num_integer_boxes:NnnNN
2001              \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
2002              \amc_num_expo_sign_int}
2003            } {}
2004          }}
2005        } {
2006          \amc_num_integer_boxes:NnnNN
2007          \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
2008          \amc_num_sign_int
2009          \int_compare:nNnTF \amc_num_expo_int > 0 {
2010            \hspace{\AMCnumeric@Hspace}\AMCexponent\hspace{\AMCnumeric@Hspace}
2011            \amc_num_integer_boxes:NnnNN
2012            \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
2013            \amc_num_expo_sign_int
2014          } {}
2015        }
2016      }
2017    }
2018  \fi

```

And tell AMC that we finished with this question:

```

2019  \ifAMC@ensemble\else\vspace{\AMCpostNquest}\par\fi
2020  \ifAMC@ensemble\ifAMCformulaire@dedans
2021    \AMCmessage{FQ}
2022  \fi\fi
2023  }
2024 }
2025

```

```

2026 \cs_new_eq:NN \AMCnumericShow \amc_numeric_show:nn
2027

```

`\AMCnumericHide` is called when the boxes are not to be drawn (in the question sheets for separate answer sheet layout), and `\AMCnumericChoices{⟨value⟩}{⟨options⟩}` is the function to be used in the LaTeX source code of the exam.

```

2028 \cs_new:Npn \amc_numeric_hide:nn #1 #2 {
2029   \keys_set:nn { amcnumeric } { #2 }
2030   \AMCncontextGoto
2031   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi
2032 }
2033
2034 \cs_new_eq:NN \AMCnumericHide \amc_numeric_hide:nn
2035
2036 \ExplSyntaxOff
2037 \def\AMCnumericChoicesPlain{%
2038   \AMC@if@separate@question{\AMC@mem@category{numeric}}{%
2039     \AMCformatChoices{\AMCnumericShow}{\AMCnumericHide}%
2040   }

```

The `{⟨value⟩}` argument is often given as a macro, that is to be expanded before calling `\AMCnumericChoicesPlain`, so that its value will be the same in the separate answer sheet...

```

2041 \ExplSyntaxOn
2042
2043 \cs_new:Npn \amc_numeric_choices:nn #1#2 {
2044   \AMCnumericChoicesPlain{#1}{#2}
2045 }
2046 \cs_generate_variant:Nn \amc_numeric_choices:nn { xn }
2047 \cs_new_eq:NN \AMCnumericChoices \amc_numeric_choices:xn
2048
2049 \ExplSyntaxOff

```

### 4.13.3 Intervals

`\AMCIntervals` The command `\AMCIntervals{⟨x⟩}{⟨x0⟩}{⟨x1⟩}{⟨delta⟩}` can be used to present answers as intervals  $[x_i, x_i + \delta[$  covering  $[\langle x0 \rangle, \langle x1 \rangle]$ , such that the only interval containing  $\langle x \rangle$  is declared as `\correctchoice`, and the other as `\wrongchoice`.

For this command to work, one has to load package `fp`.

As an example,

```

\begin{question}{quarter}
  In which interval falls  $1/4$ ?
  \begin{multicols}{5}
    \begin{choices}[o]
      \AMCIntervals{0.25}{0}{1}{0.1}
    \end{choices}
  \end{multicols}
\end{question}

```

produces (in correction mode):

**Question 12** In which interval falls  $1/4$ ?

<input type="checkbox"/> [0, 0.1[	<input checked="" type="checkbox"/> [0.2, 0.3[	<input type="checkbox"/> [0.4, 0.5[	<input type="checkbox"/> [0.6, 0.7[	<input type="checkbox"/> [0.8, 0.9[
<input type="checkbox"/> [0.1, 0.2[	<input type="checkbox"/> [0.3, 0.4[	<input type="checkbox"/> [0.5, 0.6[	<input type="checkbox"/> [0.7, 0.8[	<input type="checkbox"/> [0.9, 1[

Note that the interval formatting can be changed redefining the `\AMCintervalFormat` command, which is originally defined as

```
2050 \def\AMCintervalFormat#1#2{[#1,\,#2]}
```

to follow local conventions (writting  $[a, b]$  instead of  $[a, b[$  is for example a common usage).

```
2051 \ExplSyntaxOn
```

```
2052
```

```
2053 \fp_new:N \amc_interv_a
```

```
2054 \fp_new:N \amc_interv_b
```

```
2055 \cs_new:Npn \amc_intervals:nnnn #1 #2 #3 #4 {
```

```
2056   \fp_set:Nn \amc_interv_a { #2 }
```

```
2057   \fp_do_while:nn { \amc_interv_a < #3 } {
```

```
2058     \fp_set:Nn \amc_interv_b { \amc_interv_a + #4 }
```

```
2059     \fp_compare:nTF { \amc_interv_a <= #1 < \amc_interv_b }
```

```
2060       \correctchoice \wrongchoice
```

```
2061       {\AMCintervalFormat{\fp_use:N \amc_interv_a}{\fp_use:N \amc_interv_b}}
```

```
2062       \fp_set:Nn \amc_interv_a \amc_interv_b
```

```
2063   }
```

```
2064 }
```

```
2065 \cs_new_eq:NN \AMCIntervals \amc_intervals:nnnn
```

```
2066
```

```
2067 \ExplSyntaxOff
```

## 4.14 Open questions

`\AMCOpen` The command `\AMCOpen{<options>}{<choices>}` can be used as a replacement for the `choices` environment when asking the student to write some answer by hand. The teacher will correct and mark this answer either on the paper before scanning, or with manual data capture, thanks to the scoring boxes.

As an example,

```
\begin{question}{Linux}
```

```
  What is the first name of the person who started working on the Linux kernel?
```

```
  \AMCOpen{}{\wrongchoice[w]{w}\scoring{0}\correctchoice[c]{c}\scoring{2}}
```

```
\end{question}
```

shows:

**Question 13**    What is the first name of the person who started working on the Linux kernel?

☐ w   ☐ c

.....

The teacher will have to tick the ‘w’ box for wrong answers, and the ‘c’ box for correct answers.

Begin with the options definitions:

```

2068 \def\AMCotextGoto{}
2069 \def\AMCotextReserved{}
2070 \def\AMCocol@Background{lightgray}
2071 \def\AMCocol@BoxFrameRule{white}
2072 \def\AMCocol@FrameRule{black}
2073 \def\AMCocol@Foreground{}
2074 \def\AMCopen@answer{}
2075 \def\AMCopen@question{}
2076 \def\AMCopen@lineuptext{}
2077 \define@key{AMCOpen}{backgroundcol}{\def\AMCocol@Background{#1}}
2078 \define@key{AMCOpen}{foregroundcol}{\def\AMCocol@Foreground{#1}}
2079 \define@key{AMCOpen}{Treserved}{\def\AMCotextReserved{#1}}
2080 \define@key{AMCOpen}{question}{\AMCid@name}{\def\AMCopen@question{#1}}
2081 \define@key{AMCOpen}{answer}{\def\AMCopen@answer{#1}}
2082 \define@key{AMCOpen}{contentcommand}{\AMCopen@lines}{\def\AMCopen@contentcommand{#1}}
2083 \newdimen\AMCopen@Hspace\AMCopen@Hspace=.5em
2084 \define@key{AMCOpen}{hspace}{\AMCopen@Hspace=#1}
2085 \def\AMCopen@Width{.95\linewidth}
2086 \define@key{AMCOpen}{width}{\def\AMCopen@Width{#1}}
2087 \newdimen\AMCopen@LineHeight\AMCopen@LineHeight=1cm
2088 \define@key{AMCOpen}{lineheight}{\AMCopen@LineHeight=#1}
2089 \newcount\AMCopen@Lines\AMCopen@Lines=1
2090 \define@key{AMCOpen}{lines}{\AMCopen@Lines=#1}
2091 \newdimen\AMCopen@boxmargin\AMCopen@boxmargin=3pt
2092 \define@key{AMCOpen}{boxmargin}{\AMCopen@boxmargin=#1}
2093 \newdimen\AMCopen@boxframerule\AMCopen@boxframerule=1pt
2094 \define@key{AMCOpen}{boxframerule}{\AMCopen@boxframerule=#1}
2095 \define@key{AMCOpen}{boxframerulecol}{\def\AMCocol@BoxFrameRule{#1}}
2096 \define@key{AMCOpen}{framerulecol}{\def\AMCocol@FrameRule{#1}}
2097 \newdimen\AMCopen@framerule\AMCopen@framerule=1pt
2098 \define@key{AMCOpen}{framerule}{\AMCopen@framerule=#1}
2099 \define@key{AMCOpen}{lineuptext}{\def\AMCopen@lineuptext{#1}}
2100 \define@boolkey{AMCOpen}{dots}[true]{}
2101 \define@boolkey{AMCOpen}{scan}[true]{}
2102 \define@boolkey{AMCOpen}{retick}[true]{}
2103 \define@boolkey{AMCOpen}{annotate}[false]{}
2104 \define@boolkey{AMCOpen}{lineup}[false]{}
2105 \setkeys{AMCOpen}{dots,scan,retick,annotate,lineup,contentcommand}
2106 \newcommand\AMCopenOpts[1]{\setkeys{AMCOpen}{#1}}

```

The command \AMCOpen is similar to \AMCnumericChoices, calling either \AMCopenShow or \AMCopenHide.

```

2107 \newcommand\AMCopen@lines{%
2108   \begin{minipage}{\AMCopen@Width}%
2109     \loop\vspace{\AMCopen@LineHeight}
2110     \hspace*{.5em}\ifAMC@correc\smash{\AMCopen@answer}\def\AMCopen@answer{}\fi%
2111     \ifKV@AMCopen@dots%
2112       \dotfill\hspace*{.5em}
2113     \fi
2114     \ifnum\AMCopen@Lines>\@ne\par\advance\AMCopen@Lines\@ne\repeat%
2115   \end{minipage}
2116 }
2117 \newcommand\AMCopenShow[2]{
2118   \ifAMC@ensemble\ifAMC@formulaire@dedans%

```

```

2119 \AMCmessage{Q=\the\AMCid@quest}%
2120 \fi\fi%
2121 {\setkeys{AMCOpen}{#1}%
2122 \ifKV@AMCOpen@lineup%
2123 \ifAMC@ensemble\else%
2124 \ifx\@empty\AMCopen@lineuptext\@empty\fi%
2125 \fi%
2126 \ifAMC@correc\smash{\AMCopen@answer}\fi%
2127 \ifx\@empty\AMCopen@lineuptext\@empty%
2128 \dotfill%
2129 \else%
2130 \AMCopen@lineuptext\hfill%
2131 \fi%
2132 \else%
2133 \hspace*{.5em}\linebreak[1]\hspace*{\fill}%
2134 \fi%
2135 {\AMCnoCompleteMulti%
2136 \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2137 \def\AMCanswer##1##2{\ifAMC@ensemble ##1\else%
2138 \ifAMC@inside@box ##1\else\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2139 \hspace{\AMCopen@Hspace}}%
2140 \fboxsep=\AMCopen@boxmargin%
2141 \fboxrule=\AMCopen@boxframerule%
2142 \fcolorbox{\AMCocol@BoxFrameRule}{\AMCocol@Background}{%
2143 \ifAMC@ensemble\AMCopen@question%
2144 \ifx\@empty\AMCopen@question\@empty\else\hspace{\AMCopen@Hspace}\fi%
2145 \fi%
2146 \begin{choicescustom}[o]%
2147 \ifx\AMCocol@Foreground\@empty\@empty\else%
2148 \def\AMC@boxcolor{\AMCocol@Foreground}%
2149 \fi%
2150 #2%
2151 \ifKV@AMCOpen@scan\else\AMCdontScan\fi%
2152 \ifKV@AMCOpen@retick\AMCreTick\fi%
2153 \ifKV@AMCOpen@annotate\else\AMCdontAnnotate\fi%
2154 \end{choicescustom}}%
2155 \ifx\@empty\AMCotextReserved\@empty%
2156 \hspace{-\AMCopen@Hspace}%
2157 \else%
2158 \ifx\AMCocol@Foreground\@empty\@empty%
2159 \AMCotextReserved%
2160 \else%
2161 \textcolor{\AMCocol@Foreground}{\AMCotextReserved}%
2162 \fi%
2163 \fi%
2164 }}%
2165 \ifKV@AMCOpen@lineup\else%
2166 \par\nobreak\noindent%
2167 \hspace*{\fill}{%
2168 \fboxrule=\AMCopen@framerule%
2169 \fcolorbox{\AMCocol@FrameRule}{white}{%
2170 \csname\AMCopen@contentcommand\endcsname
2171 }}%

```

```

2172     \vspace{\AMCpostQuest}\par%
2173     \fi%
2174 }%
2175 \ifAMC@ensemble\ifAMCformulaire@dedans%
2176 \AMCmessage{FQ}%
2177 \fi\fi%
2178 }
2179 \newcommand\AMCOpenHide[2]{%
2180 \AMCotextGoto%
2181 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2182 }
2183 \def\AMCOpen{%
2184 \AMC@if@separate@question{\AMC@mem@category{open}}%
2185 \AMCformatChoices{\AMCOpenShow}{\AMCOpenHide}%
2186 }

```

#### 4.15 Boxes with letters only

`\AMCBoxOnly` Sometimes the letters printed in the boxes (or just after them) are enough to describe the answers. In such cases, printing the boxes both on the question and on the answer sheet is not necessary. The `\AMCBoxOnly{<options>}{<choices>}` can be used as a replacement for the `choices` environment:

```

\begin{question}{arm}
  Which letter shows the \textit{arm} on the diagram?
  \AMCBoxOnly{ordered=true}{\wrongchoice[A]{}\correctchoice[B]{}%
    \wrongchoice[C]{}\wrongchoice[D]{}%
\end{question}

2187 \def\AMCbotextGoto{}
2188 \def\AMCbo@help{}
2189 \define@key{AMCBoxOnly}{help}{\def\AMCbo@help{#1}}
2190 \define@boolkey{AMCBoxOnly}{ordered}[false]{}
2191 \setkeys{AMCBoxOnly}{ordered}
2192 \newcommand\AMCboOpts[1]{\setkeys{AMCBoxOnly}{#1}}
2193 \newcommand\AMCboShow[2]{%
2194 \ifAMC@ensemble\ifAMCformulaire@dedans%
2195 \AMCmessage{Q=\the\AMCid@quest}%
2196 \fi\fi%
2197 {\setkeys{AMCBoxOnly}{#1}%
2198 \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2199 \def\AMCanswer##1##2{\hspace{\AMCformHSpace} \ifAMC@ensemble ##1\else%
2200 \ifAMC@inside@box ##1\else\AMCboxOutsideLetter{##1}{##2}\fi\fi%
2201 }%
2202 \ifAMC@ensemble\AMCbo@help\fi%
2203 \ifKV@AMCBoxOnly@ordered%
2204 \begin{choicescustom}[o]%
2205 \else%
2206 \begin{choicescustom}%
2207 \fi%
2208 #2
2209 \end{choicescustom}%
2210 }%

```

```

2211 \ifAMC@ensemble\ifAMCformulaire@dedans%
2212 \AMCmessage{FQ}%
2213 \fi\fi%
2214 }
2215 \newcommand\AMCboHide[2]{
2216 \AMCbotextGoto%
2217 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2218 }
2219 \def\AMCBoxOnly{%
2220 \AMC@if@separate@question{\AMC@mem@category{box}}}%
2221 \AMCformatChoices{\AMCboShow}{\AMCboHide}%
2222 }

```

## 4.16 Page formatting

### 4.16.1 Watermark

`\AMCw@termark` These commands are used to print a grey “DRAFT” under each page, so as to prevent from  
`\AMCw@terprint` printing old versions of the subject.

```

2223 \DeclareFontShape{OT1}{cmr}{b}{n}{<35->cmr17}{-}
2224 \def\AMC@watertext{\AMC@loc@draft}
2225 \newcommand\AMCw@termark{%
2226 \setlength{\@tempdimb}{.5\paperwidth}%
2227 \setlength{\@tempdimc}{-.5\paperheight}%
2228 \put(\strip@pt\@tempdimb,\strip@pt\@tempdimc){%
2229 \makebox(0,0){\rotatebox{45}{\AMC@LR{%
2230 \textcolor{gray}{0.8}{
2231 \fontencoding{OT1}\fontfamily{cmr}
2232 \fontseries{b}\fontshape{n}
2233 \fontsize{90pt}{120pt}
2234 \selectfont
2235 \AMC@watertext}}}}}}
2236 \newcommand\AMCw@terprint[1]{%
2237 \setbox\@tempboxa\vbox to \z@{%
2238 \vbox{%
2239 \hbox to \z@{%
2240 #1\hss}}\vss}
2241 \dp\@tempboxa\z@
2242 \box\@tempboxa}

```

### 4.16.2 Signs for scan analysis

The following code sets up all the signs to be printed on the pages so as to be able to recognize the position of the boxes on the scans. Four circles ● are printed on the corners (see `\m@rqueCalage`), and binary boxes show the student sheet number (see `\AMCIDBoxesA`), the page (see `\AMCIDBoxesB`) and a checking number (see `\AMCIDBoxesC`).

`\AMC@intituleHead` is the title to be printed at the beginning (used for corrected sheet, and empty on subject). `\AMC@note` is printed at the bottom of each page. You can change its value using `\AMCsetFoot{<foot>}`.

```

2243 \def\AMCccercle#1#2{%
2244 {\setlength{\unitlength}{1mm}%
2245 \begin{picture}(\#1,\#1)(-\#2,-\#2)\thinlines\circle*{\#1}\end{picture}}

```



```

2246 \def\m@rqueCalage{\AMC@circle{3.6}{1.8}}
2247 \def\m@rque#1{\AMC@tracebox{1}{#1}{\m@rqueCalage}}
2248 \def\he@dtaille#1{% \par cancels the \leavevmode
2249 % introduced by https://github.com/pietvo/fancyhdr/commit/6b1ad10eeb5bc3d804f3cd2cf193e6440d0229e6
2250 \par\vbox to 1cm{#1}}
2251 \def\he@dbas#1{\he@dtaille{\vspace*{\fill}#1}}
2252 \def\he@dhaut#1{\he@dtaille{#1\vspace*{\fill}}}}
2253 \def\AMC@intituleHead{\AMC@loc@corrected}
2254 \def\AMC@note{}
2255 \def\AMCsetFoot#1{\def\AMC@note{#1}}
2256 \newcommand\AMCStudentNumber{\the\AMCid@etud}
2257 \newcommand\AMCIDBoxesA{\AMC@binaryCode{id=1,ndigits=\AMC@NCBetud}{\the\AMCid@etud}}
2258 \newcommand\AMCIDBoxesB{\AMC@binaryCode{id=2,ndigits=\AMC@NCBpage}{\thepage}}
2259 \newcommand\AMCIDBoxesC{\AMC@binaryCode{id=3,ndigits=\AMC@NCBcheck}{\the\AMCid@check}}
2260 \newcommand\AMCIDBoxesABC{%
2261 \hbox{\vbox{\noindent\AMCIDBoxesA\
2262 \noindent\AMCIDBoxesB\AMCIDBoxesC}}}%
2263 }
2264 \AtBeginPage{\ifAMC@pagelayout\global\advance\AMCid@check\m@ne%
2265 \ifnum\AMCid@check<1\global\AMCid@check=\AMCid@checkmax\fi%
2266 \AMC@pagepos%
2267 \ifAMC@watermark\ifAMC@correthead\else\AMCw@terprint{\AMCw@termark}%
2268 \fi\fi\fi}
2269 \fancypagestyle{AMCpageHeadOnly}{%
2270 \fancyhf{} \fancyhead[C]{\textsc{\AMC@intituleHead}}%
2271 \renewcommand{\headrulewidth}{0pt}%
2272 \renewcommand{\footrulewidth}{0pt}%
2273 }
2274 \fancypagestyle{AMCpageFull}{%
2275 \fancyhf{}%
2276 \fancyhead[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHG}}}}%
2277 \fancyhead[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHD}}}}%
2278 \fancyfoot[L]{\AMC@LR{\leavevmode\m@rque{positionBG}}}%
2279 \fancyfoot[R]{\AMC@LR{\leavevmode\m@rque{positionBD}}}%
2280 \fancyhead[C]{\AMC@LR{\he@dhaut{%
2281 \begin{minipage}[b]{\AMC@CBtaille}\AMCboxColor{black}%
2282 \ifAMCids@top\vbox to \AMCids@height{\texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2283 \AMCIDBoxesABC
2284 \end{minipage}%
2285 \ifAMCids@side\hbox to \AMCids@width{\hspace*{\fill}%
2286 \texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2287 }}}}%
2288 \fancyhfoffset[EOLR]{5mm}%
2289 \fancyfoot[C]{\AMC@note}%
2290 \renewcommand{\headrulewidth}{0pt}%
2291 \renewcommand{\footrulewidth}{0pt}%
2292 }
2293 \newcommand\AMCsubjectPageTag{%
2294 \fbox{\texttt{\the\AMCid@etud:\thepage}}%
2295 }
2296 \fancypagestyle{AMCpageNoMarks}{%
2297 \fancyhf{}%
2298 \fancyhead[R]{\AMCsubjectPageTag}%

```

```

2299 \fancyfoot[C]{\AMC@note}%
2300 \renewcommand{\headrulewidth}{0pt}%
2301 \renewcommand{\footrulewidth}{0pt}%
2302 }
2303 \fancypagestyle{AMCpageEmpty}{%
2304 \fancyhf{}%
2305 \renewcommand{\headrulewidth}{0pt}%
2306 \renewcommand{\footrulewidth}{0pt}%
2307 }
2308 \AtBeginDocument{%
2309 \ifAMC@pagelayout%
2310 \ifAMC@correthead
2311 \pagestyle{AMCpageHeadOnly}
2312 \else
2313 \pagestyle{AMCpageFull}
2314 \fi
2315 \fi
2316 }

```

## 4.17 Defining a single exam copy content

`\onecopy` The command `\onecopy[ $\langle n \rangle$ ]{ $\langle code \rangle$ }` generates  $\langle n \rangle$  copies of the subject that is described in  $\langle code \rangle$ . The L<sup>A</sup>T<sub>E</sub>X code  $\langle code \rangle$  that generates a single copy can be a little long, so that the environment `examcopy` is often preferred.

```

2317 \newcommand{\onecopy}[2]{%
2318 \ifx\AMCNombreCopies\undefined\AMCnum@copies=#1%
2319 \else\AMCnum@copies=\AMCNombreCopies\fi%
2320 \AMCmessage{TOTAL=\the\AMCnum@copies}%
2321 \message{^^JAMC:copies:total=\the\AMCnum@copies^^J}%
2322 \AMCid@etud=\AMCid@etudstart%
2323 \ifnum\AMCid@etud=0\AMCid@etud=\AMC@premierecopie\fi%
2324 \AMCid@etudfin=\AMCnum@copies%
2325 \advance\AMCid@etudfin\AMCid@etud\relax%
2326 \ifAMC@correthead\AMCid@etudfin=\AMC@premierecopie%
2327 \message{^^JAMC:copies:total=1^^J}%
2328 \fi
2329 \ifAMC@pdfform\begin{Form}\fi%
2330 \loop{%
2331 \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
2332 \string\rngstate{\the\AMCid@etud}{\the\AMC@SR}%
2333 }\fi%
2334 \AMC@zoneformulairefalse\setcounter{page}{1}\setcounter{section}{0}%
2335 \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageNoMarks}\fi\fi%
2336 \AMCnumero{1}%
2337 \ifAMC@calibration\AMCmessage{ETU=\the\AMCid@etud}\fi%
2338 \AMC@multiclear%
2339 \global\AMC@keepmemoryfalse%
2340 #2%
2341 \ifAMC@keepmemory\else\AMC@mem@clear\fi%
2342 \clearpage}%
2343 \message{^^JAMC:copies:add=1^^J}%
2344 \advance\AMCid@etud\@ne\ifnum\AMCid@etud<\AMCid@etudfin\repeat%
2345 \global\AMCid@etudstart=\AMCid@etud%

```

```

2346 \ifAMC@pdfform\end{Form}\fi%
2347 \AMC@multi@report%
2348 }

```

**\AMCaddpagesto** In some situations, one needs all question sheets to have the same number of pages. The command **\AMCaddpagesto{<n>}** adds enough (white) pages to get at least <n> pages in the current question sheet.

```

2349 \newcount\AMC@addpages
2350 \newcommand{\AMCaddpagesto}[1]{%
2351   \AMC@addpages=#1\advance\AMC@addpages\@ne%
2352   \clearpage%
2353   \@whilenum\thepage<\AMC@addpages\do{%
2354     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2355     \hbox{}\clearpage%
2356   }%
2357 }

```

**\AMCcleardoublepage** If you want to print the subject all at one time in duplex mode, it is necessary to end each subject with an even number of pages. This can be achieved using **\AMCcleardoublepage** at the end of the copy definition. This command is also useful inserted before the separate answer sheet (if any).

```

2358 \def\AMCcleardoublepage{%
2359   \clearpage%
2360   \ifodd\thepage\else%
2361     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2362     \hbox{}\clearpage%
2363   \fi%
2364 }

```

**\exemplairepair** To make some differences in the copies, checking if the student sheet number is odd, with **\exemplairepair** construct, can be useful.

```

2365 \def\exemplairepair{\ifodd\AMCid@etud}

```

**\AMClabel** Commands **\AMClabel**, **\AMCref** and **\AMCpageref** replaces L<sup>A</sup>T<sub>E</sub>X's **\label**, **\ref** and **\pageref** **\AMCref** to be able to use different labels for different sheets.

```

\AMCref
2366 \newcommand\AMCstudentlabel[1]{\the\AMCid@etud-#1}
2367 \def\AMClabel#1{\expandafter\label{\AMCstudentlabel{#1}}}
2368 \def\AMCref#1{\expandafter\ref{\AMCstudentlabel{#1}}}
2369 \def\AMCpageref#1{\expandafter\pageref{\AMCstudentlabel{#1}}}

```

**\AMCqlabel** A label can be created for current question with **\AMCqlabel{<label>}**. This label can be used with **\AMCref** and **\AMCpageref**. This command is defined for backward compatibility only, since **\AMClabel** can also be used.

```

2370 \newcommand{\AMCqlabel}[1]{%
2371   \AMClabel{#1}%
2372 }

```

## 4.18 Pre-association

**\AMCassociation** Association between sheets and students can be made before the exam with the **\AMCassociation[<filename>]{<id>}** command. The optional argument <filename> will be used when printing student sheets to files.

```

2373 \newcommand{\AMCassociation}[2][]{%
2374   \ifAMC@calibration%
2375   \immediate\write\AMC@XYFILE{\string\association{\the\AMCid@etud}{#2}{#1}}%
2376   \fi%
2377 }

```

**AMCstudentslistfile** You can also pass AMC the path to the CSV file with students, and the unique key that can be used, with `\AMCstudentslistfile{<path>}{<key>}`.

```

2378 \newcommand{\AMCstudentslistfile}[2]{%
2379   \ifAMC@calibration%
2380   \immediate\write\AMC@XYFILE{\string\with{studentslistfile=#1}}%
2381   \immediate\write\AMC@XYFILE{\string\with{studentslistkey=#2}}%
2382   \fi%
2383 }

```

## 4.19 Package options

See section 3.1 for the options descriptions.

```

2384 \def\AMC@lang@code{}
2385 \DeclareOptionX{noshuffle}{\AMC@ordretrue}
2386 \DeclareOptionX{noshufflegroups}{\AMC@shuffleGfalse}
2387 \DeclareOptionX{fullgroups}{\AMC@fullGroupstrue}
2388 \DeclareOptionX{answers}{\AMC@corretheadtrue\AMC@correcttrue}
2389 \DeclareOptionX{indivanswers}{\AMC@correcttrue}
2390 \DeclareOptionX{textpos}{\AMC@textPostrue}
2391 \DeclareOptionX{extractonly}{\AMC@extractOnlytrue\AMC@textPostrue\AMC@boxStyle{shape=none}\AMC@boxedAnswer}
2392 \DeclareOptionX{box}{\AMC@qbloctrue}
2393 \DeclareOptionX{asbox}{\AMC@asqbloctrue}
2394 \DeclareOptionX{separateanswersheet}{\AMC@ensembletrue}
2395 \DeclareOptionX{digits}{\AMC@inside@digittrue}
2396 \DeclareOptionX{ordre}{\AMC@ordretrue}
2397 \DeclareOptionX{correc}{\AMC@corretheadtrue\AMC@correcttrue}
2398 \DeclareOptionX{modele}{\AMC@corretheadtrue\AMC@correcfalse\AMC@ordretrue}
2399 \DeclareOptionX{correcindiv}{\AMC@correcttrue}
2400 \DeclareOptionX{init}{\AMC@SR@time}
2401 \DeclareOptionX{bloc}{\AMC@qbloctrue}
2402 \DeclareOptionX{completemulti}{\AMC@complete@multitrue}
2403 \DeclareOptionX{insidebox}{\AMC@inside@boxtrue}
2404 \DeclareOptionX{ensemble}{\AMC@ensembletrue}
2405 \DeclareOptionX{chiffres}{\AMC@inside@digittrue}
2406 \DeclareOptionX{outsidebox}{\AMC@outside@boxtrue}
2407 \DeclareOptionX{calibration}{\AMC@calibrationtrue}
2408 \DeclareOptionX{nowatermark}{\AMC@watermarkfalse}
2409 \newcommand\AMC@catalogMode{%
2410   \AMC@catalogtrue%
2411   \AMC@watermarkfalse\AMC@corretheadtrue%
2412   \AMC@correcttrue\AMC@ordretrue\AMC@shuffleGfalse%
2413   \AMC@fullGroupstrue%
2414   \def\AMC@intituleHead{\AMC@loc@catalog}\AMC@affichekeystrue}
2415 \newcommand\AMC@keys@next{\AMC@keyslinefalse}
2416 \newcommand\AMC@keys@line{\AMC@keyslinetrue}
2417 \DeclareOptionX{catalog}{\AMC@catalogMode}
2418 \DeclareOptionX{keys}[next]{\csname AMC@keys@#1\endcsname{}}

```

```

2419 \DeclareOptionX{francais}{\def\AMC@lang@code{FR}\AMC@loc@FR}
2420 \DeclareOptionX{lang}{\def\AMC@lang@code{#1}\csname AMC@loc@#1\endcsname}
2421 \DeclareOptionX{versionA}{%
2422   \def\AMCid@checkmax{31}\def\AMC@NCBetud{9}\def\AMC@NCBpage{4}%
2423   \def\AMC@NCBcheck{5}\setlength{\AMC@CBtaille}{4cm}%
2424   \def\AMC@premierecopie{100}}
2425 \DeclareOptionX{plain}{\AMC@plaintrue}
2426 \DeclareOptionX{nopage}{\AMC@pagelayoutfalse}
2427 \DeclareOptionX{postcorrect}{\AMC@postcorrecttrue}
2428 \DeclareOptionX{automarks}{\AMC@automarkstrue}
2429 \newif\ifAMCneeds@storebox\AMCneeds@storeboxfalse
2430 \DeclareOptionX{storebox}{\AMCneeds@storeboxtrue}
2431 \DeclareOptionX{pdfform}{\AMC@pdfformtrue}
2432 \DeclareOptionX{codedigit}{\AMCcodeID@{#1}}
2433 \newif\ifAMC@survey\AMC@surveyfalse
2434 \DeclareOptionX{survey}{\AMC@surveytrue}
2435 \ProcessOptionsX
2436 \ifAMCneeds@storebox
2437   \RequirePackage{storebox}\AtBeginDocument{{}}%
2438 \fi
2439 \ifAMC@pdfform
2440   \AMCmessage{VAR:project:pdfform=1}%
2441   \AMCboxStyle{shape=form}%
2442   \RequirePackage[pageanchor=false]{hyperref}%
2443 \else%
2444   \AMCmessage{VAR:project:pdfform=0}%
2445 \fi
2446 \AtBeginDocument{%
2447   \ifAMCneeds@storebox%
2448     \let\AMC@new@savebox=\newstorebox%
2449     \let\AMC@save@box=\storebox%
2450     \let\AMC@use@box=\usestorebox%
2451   \fi%
2452   \AMC@new@savebox{\AMC@ovalbox@R}%
2453   \AMC@new@savebox{\AMC@ovalbox@RF}%
2454   \AMC@new@savebox{\AMC@ovalbox@}%
2455   \AMC@new@savebox{\AMC@ovalbox@F}%
2456   \AMC@shapeprepare%
2457 }

```

## 4.20 Survey add-on

Some code and *tikz* settings to help handling surveys, see [https://survey.codes/pdf/surveyamc\\_manual.pdf](https://survey.codes/pdf/surveyamc_manual.pdf) for more details. This survey add-on is originally written by Claudia Saalbach.

```

questionnaires (env.)
  auto (env.) 2458 \ifAMC@survey
question-auto (env.) 2459 \NewEnviron{Questionnaires}[1]{
  values (env.) 2460 \onecopy{#1}{
  values-auto (env.) 2461 \BODY
variable-auto (env.) 2462 }
  \answer 2463 }

```

```

2464 \NewEnviron{auto}[1]{
2465 \csvreader[head to column names, separator=tab]{#1}{}{
2466 \BODY
2467 }
2468 }
2469 \NewEnviron{question-auto}[3]{
2470 \csvreader[head to column names, separator=tab]{#1}{}{
2471 \ifcsvstrcmp{#2}{#3}{\BODY \\\}{}
2472 }
2473 }
2474 \newenvironment{values}{}{}
2475 \NewEnviron{values-auto}[5]{
2476 \csvreader[head to column names, separator=tab]{#1}{}{
2477 \ifcsvstrcmp{#2}{#3}{
2478 \ifcsvstrcmp{#4}{#5}{\BODY \\\
2479 }{}
2480 }{}
2481 }
2482 }
2483 \NewEnviron{variable-auto}[3]{
2484 \foreach \x in {#3}{
2485 \csvreader[head to column names, separator=tab]{#1}{}{
2486 \ifcsvstrcmp{#2}{\x}{\BODY}{}
2487 }
2488 }
2489 }
2490 \newcommand{\answer}[5][\global\advance\AMCrep@count\@ne\relax%
2491 \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%
2492 \global\AMCune@bonnettrue%
2493 \AMCload@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkbox}%
2494 \else\AMC@box{#1}{\fi}{#2}{#3}{#4}{#5}}{\the\AMCrep@count}\ignorespaces}
2495 \RequirePackage{tikz}
2496 \usetikzlibrary{positioning, shapes, arrows, tikzmark, decorations.pathreplacing}
2497 \tikzset{
2498   checkbox-sc/.style={
2499     right=of lab\thecsvrow
2500   },
2501   vallab-sc/.style={
2502     text width=4cm,
2503     align=left,
2504   },
2505   checkbox-mc/.style={
2506   },
2507   vallab-mc/.style={
2508     above=of box\thecsvrow,
2509     text width=1.4cm,
2510     align=center,
2511   },
2512   varlab-mc/.style={
2513     text width=4cm,
2514     align=left,
2515   },
2516   node distance= 0mm,

```

```

2517 }
2518 \fi

```

## 4.21 Package Errors

`\AMC@error@explain` Error to display if `\explain` command is used outside question like environments

```

2519 \def\AMC@error@explain{\PackageError{automultiplechoice}{
2520     Command \protect\explain\space can only be used inside\MessageBreak question like environments}{Some
2521 }}

```

## 4.22 Optional features

This package tries to see if optional packages `environ` and `etex` are loadable, and load them if possible. This behaviour can be cancelled by using `plain` option.

```

2522 \ifAMC@plain
2523 \else
2524     \IfFileExists{environ.sty}{\RequirePackage{environ}}{}
2525     \ifx\TeXversion\@undefined
2526     \else
2527         \RequirePackage{etex}
2528     \fi
2529 \fi

```

`examcopy` (*env.*) Then, if `environ` package is loaded and defines command `\NewEnviron`, environment `examcopy` is defined.

Environment `{examcopy}[\langle n \rangle]` does the same as command `onecopy`: it encloses L<sup>A</sup>T<sub>E</sub>X code which makes *one* exam copy. Optional argument  $\langle n \rangle$  gives the number of desired copies – this can also be modified redefinig `\AMCNombreCopies`.

```

2530 \@ifpackageloaded{environ}{%
2531     \ifx\NewEnviron\undefined\PackageWarning{automultiplechoice}%
2532     {Package environ loaded but too old version:
2533         environnement examcopy/copieexamen will NOT be defined.}%
2534     \else\NewEnviron{examcopy}[1][5]{\onecopy{#1}{\BODY}}\fi}%
2535 {\PackageWarning{automultiplechoice}%
2536     {Package environ not loaded: environnement
2537         examcopy/copieexamen will NOT be defined.}}

```

## 4.23 Use with recent LuaTeX versions

In recent LuaTeX versions, the commands `pdfsavepos`, `pdflastxpos` and `pdflastypos` has been renamed, stripping the `pdf` part. The following code tries to detect this situation and make the bindings between the old and new command names.

```

2538 \ExplSyntaxOn
2539
2540 \cs_if_exist:NTF \pdfsavepos { } {
2541     \cs_if_exist:NTF \savepos { \cs_new_eq:NN \pdfsavepos \savepos } { }
2542 }
2543 \cs_if_exist:NTF \pdflastxpos { } {
2544     \cs_if_exist:NTF \lastxpos { \cs_new_eq:NN \pdflastxpos \lastxpos } { }
2545 }
2546 \cs_if_exist:NTF \pdflastypos { } {

```

```

2547 \cs_if_exist:NTF \lastypos { \cs_new_eq:NN \pdflastypos \lastypos } { }
2548 }

```

In some situations, the *page* dimensions are different from the *paper* dimensions. This must be taken into account when computing coordinates.

```

2549
2550 \cs_if_exist:NTF \pdfpagewidth { } {
2551   \cs_new_eq:NN \pdfpagewidth \paperwidth
2552 }
2553 \cs_if_exist:NTF \pdfpageheight { } {
2554   \cs_new_eq:NN \pdfpageheight \paperheight
2555 }
2556
2557 \ExplSyntaxOff

```

## 4.24 External control

Some of the package options can be controlled defining `\xxxExterne` commands. For example, the following command will format the subject document, whatever options are used in the L<sup>A</sup>T<sub>E</sub>X file:

```

\pdfatex '\nonstopmode\def\SujetExterne{1}\def\NoWatermarkExterne{1}\input{mcq.tex}'

```

```

2558 \ifx\SujetExterne\undefined\else
2559 \message{***SUJET***^^J}
2560 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse
2561 \fi
2562 \ifx\ScoringExterne\undefined\else
2563 \message{***SCORING***^^J}
2564 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse\AMC@invisibletrue
2565 \fi
2566 \ifx\CorrigeExterne\undefined\else
2567 \message{***CORRIGE***^^J}
2568 \AMC@calibrationfalse\AMC@corretheadtrue\AMC@correcttrue\AMC@watermarkfalse
2569 \fi
2570 \ifx\CorrigeIndivExterne\undefined\else
2571 \message{***CORRIGE***^^J}
2572 \AMC@calibrationtrue\AMC@corretheadfalse\AMC@correcttrue\AMC@watermarkfalse
2573 \fi
2574 \ifx\CatalogExterne\undefined\else
2575 \message{***CATALOG***^^J}
2576 \AMC@catalogMode
2577 \fi
2578 \ifx\NoWatermarkExterne\undefined\else
2579 \AMC@watermarkfalse
2580 \fi
2581 \ifx\codeDigitExterne\undefined\else
2582 \AMCcodeID@@{\codeDigitExterne}
2583 \fi

```

## 4.25 Page layout

The following code sets the correct page layout to have room for signs for scan analysis, and prepares watermark printing:



```

2584 \@ifpackageloaded{geometry}{\usepackage{geometry}}
2585 \ifAMC@pagelayout
2586   \ifAMC@correthead
2587     \geometry{hmargin=3cm,vmargin={1cm,1cm},includeheadfoot,headheight=1cm,footskip=1cm}
2588   \else
2589     \geometry{hmargin=3cm,headheight=2cm,headsep=.3cm,footskip=1cm,top=3.5cm,bottom=2.5cm}
2590   \fi
2591   \ifAMC@watermark
2592     \ifAMC@correthead\else
2593       \def\AMC@note{\begin{minipage}{0.65\linewidth}
2594         \AMC@LR{\textcolor{blue}{\AMC@loc@message}}
2595       \end{minipage}}
2596     }
2597   \fi
2598 \fi
2599 \fi

```

## 4.26 Initialisation

Initialisation of the check counter:

```
2600 \AMCid@check=\AMCid@checkmax\advance\AMCid@check@one
```

Telling outside if separate answer sheet, and boxes labelling, are requested:

```

2601 \ifAMC@ensemble\AMCmessage{VAR:ensemble=1}\fi
2602 \ifAMC@inside@box\AMCmessage{VAR:insidebox=1}\fi
2603 \ifAMC@outside@box\AMCmessage{VAR:outsidebox=1}\fi
2604 \ifAMC@postcorrect\AMCmessage{VAR:postcorrect=1}\fi

```

Preparing writing to .xy file :

```

2605 \ifAMC@calibration
2606 \newwrite\AMC@XYFILE%
2607 \immediate\openout\AMC@XYFILE\jobname.xy%
2608 \immediate\write\AMC@XYFILE{\string\version{\AMC@VERSION}}
2609 \immediate\write\AMC@XYFILE{\string\with{codedigit=\AMCcodeID@mode}}
2610 \immediate\write\AMC@XYFILE{\string\with{version=\AMC@VERSION}}
2611 \immediate\write\AMC@XYFILE{\string\with{ensemble=\ifAMC@ensemble yes\else no\fi}}
2612 \immediate\write\AMC@XYFILE{\string\with{insidebox=\ifAMC@inside@box yes\else no\fi}}
2613 \immediate\write\AMC@XYFILE{\string\with{outsidebox=\ifAMC@outside@box yes\else no\fi}}
2614 \immediate\write\AMC@XYFILE{\string\with{postcorrect=\ifAMC@postcorrect yes\else no\fi}}
2615 \immediate\write\AMC@XYFILE{\string\with{extractonly=\ifAMC@extractOnly yes\else no\fi}}
2616 \immediate\write\AMC@XYFILE{\string\with{lang=\AMC@lang@code}}
2617 \ifx\AMCNombreCopies\undefined%
2618 \immediate\write\AMC@XYFILE{\string\with{ncopies=default}}}%
2619 \else%
2620 \immediate\write\AMC@XYFILE{\string\with{ncopies=\AMCNombreCopies}}}%
2621 \fi%
2622 \fi

```

## 4.27 French command names

For backward compatibility, a lot of commands have their french counterpart:

```

2623 \let\reponses=\choices\let\endreponses=\endchoices
2624 \let\reponseshoriz=\choiceshoriz\let\endreponseshoriz=\endchoiceshoriz

```

```

2625 \let\reponsesperso=\choicescustom\let\endreponsesperso=\endchoicescustom
2626 \let\bonne=\correctchoice
2627 \let\mauvaise=\wrongchoice
2628 \let\bareme=\scoring
2629 \let\baremeDefautM=\scoringDefaultM
2630 \let\baremeDefautS=\scoringDefaultS
2631 \def\exemplaire{\AMC@loc@FR\onecopy}
2632 \@ifpackageloaded{environ}{%
2633   \let\copieexamen=\examcopy\let\endcopieexamen=\endexamcopy}{%
2634   \let\melangegroupe=\shufflegroup
2635   \let\restituegroupe=\insertgroup
2636   \let\alafin=\lastchoices
2637   \let\formulaire=\AMCform
2638   \let\AMCdebutFormulaire=\AMCformBegin
2639   \let\champnom=\namefield
2640   \let\choixIntervalles=\AMCIntervals

```

## 5 Outputs

In the .xy file, 0/⟨*n*⟩ means student sheet number 0 (there is only one “student sheet” numbered 0 for this document as we did not use \onecopy) and page number ⟨*n*⟩ inside this student sheet. Then, each instance of the \tracepos command shows *x* and *y* positions as arguments #2 and #3 (unit is sp, such that  $65536 \times 72.27 \text{ sp}$  is one inch). One has to take min and max of the *x*-values to determine the left and right position of the box, and min and max values of *y*-values to determine top and bottom position of the box.

### 5.1 namefield command

Lines in the .xy file from a \namefield command:

```

\tracepos{0/34: __zone: id: __n}{0sp}{25231091sp}{square}
\tracepos{0/34: __zone: id: __n}{5873801sp}{0sp}{square}
\tracepos{0/34: __zone: id: __n}{15861297sp}{0sp}{square}
\tracepos{0/34: __zone: id: __n}{0sp}{22245913sp}{square}

```

### 5.2 AMCboxedchar command

Lines in the .xy file from a \AMCboxedchar command:

```

\tracepos{0/35: test}{22855914sp}{48672399sp}{square}
\tracepos{0/35: test}{23561334sp}{47966979sp}{square}

```

### 5.3 AMCcode command

Lines in the .xy file from a \AMCcode command. Here, code[⟨*n*⟩]:⟨*q*⟩,⟨*v*⟩ relates to digit number ⟨*n*⟩ from the right (⟨*n*⟩=1 for units, ⟨*n*⟩=2 for tens, ⟨*n*⟩=3 for hundreds and so on), question number ⟨*q*⟩ (\AMCcode uses a fake question; this number can be ignored), and value ⟨*v*⟩-1 (box number ⟨*v*⟩ for the digit).

```

\tracepos{0/58: case: code[5]: 16, 1}{24875504sp}{43399484sp}{square}
\tracepos{0/58: case: code[5]: 16, 1}{25580924sp}{42694064sp}{square}

```

```

\boxchar{0/58:case:code[5]:16,1}{A}
\tracepos{0/58:case:code[5]:16,2}{24875504sp}{42285372sp}{square}
\tracepos{0/58:case:code[5]:16,2}{25580924sp}{41579952sp}{square}
\boxchar{0/58:case:code[5]:16,2}{B}
\tracepos{0/58:case:code[5]:16,3}{24875504sp}{41171260sp}{square}
\tracepos{0/58:case:code[5]:16,3}{25580924sp}{40465840sp}{square}
\boxchar{0/58:case:code[5]:16,3}{C}
\tracepos{0/58:case:code[5]:16,4}{24875504sp}{40057148sp}{square}
\tracepos{0/58:case:code[5]:16,4}{25580924sp}{39351728sp}{square}
\boxchar{0/58:case:code[5]:16,4}{D}
\tracepos{0/58:case:code[4]:17,1}{26540303sp}{45627708sp}{square}
\tracepos{0/58:case:code[4]:17,1}{27245723sp}{44922288sp}{square}
\boxchar{0/58:case:code[4]:17,1}{0}
\tracepos{0/58:case:code[4]:17,2}{26540303sp}{44513596sp}{square}
\tracepos{0/58:case:code[4]:17,2}{27245723sp}{43808176sp}{square}
\boxchar{0/58:case:code[4]:17,2}{1}
\tracepos{0/58:case:code[4]:17,3}{26540303sp}{43399484sp}{square}
\tracepos{0/58:case:code[4]:17,3}{27245723sp}{42694064sp}{square}
\boxchar{0/58:case:code[4]:17,3}{2}
\tracepos{0/58:case:code[4]:17,4}{26540303sp}{42285372sp}{square}
\tracepos{0/58:case:code[4]:17,4}{27245723sp}{41579952sp}{square}
\boxchar{0/58:case:code[4]:17,4}{3}
\tracepos{0/58:case:code[4]:17,5}{26540303sp}{41171260sp}{square}
\tracepos{0/58:case:code[4]:17,5}{27245723sp}{40465840sp}{square}
\boxchar{0/58:case:code[4]:17,5}{4}
\tracepos{0/58:case:code[4]:17,6}{26540303sp}{40057148sp}{square}
\tracepos{0/58:case:code[4]:17,6}{27245723sp}{39351728sp}{square}
\boxchar{0/58:case:code[4]:17,6}{5}
\tracepos{0/58:case:code[3]:18,1}{28032160sp}{45627708sp}{square}
\tracepos{0/58:case:code[3]:18,1}{28737580sp}{44922288sp}{square}
\boxchar{0/58:case:code[3]:18,1}{0}
\tracepos{0/58:case:code[3]:18,2}{28032160sp}{44513596sp}{square}
\tracepos{0/58:case:code[3]:18,2}{28737580sp}{43808176sp}{square}
\boxchar{0/58:case:code[3]:18,2}{1}
\tracepos{0/58:case:code[3]:18,3}{28032160sp}{43399484sp}{square}
\tracepos{0/58:case:code[3]:18,3}{28737580sp}{42694064sp}{square}
\boxchar{0/58:case:code[3]:18,3}{2}
\tracepos{0/58:case:code[3]:18,4}{28032160sp}{42285372sp}{square}
\tracepos{0/58:case:code[3]:18,4}{28737580sp}{41579952sp}{square}
\boxchar{0/58:case:code[3]:18,4}{3}
\tracepos{0/58:case:code[3]:18,5}{28032160sp}{41171260sp}{square}
\tracepos{0/58:case:code[3]:18,5}{28737580sp}{40465840sp}{square}
\boxchar{0/58:case:code[3]:18,5}{4}
\tracepos{0/58:case:code[3]:18,6}{28032160sp}{40057148sp}{square}
\tracepos{0/58:case:code[3]:18,6}{28737580sp}{39351728sp}{square}
\boxchar{0/58:case:code[3]:18,6}{5}
\tracepos{0/58:case:code[2]:19,1}{29524017sp}{45627708sp}{square}
\tracepos{0/58:case:code[2]:19,1}{30229437sp}{44922288sp}{square}
\boxchar{0/58:case:code[2]:19,1}{0}

```

```

\tracepos{0/58:case:code[2]:19,2}{29524017sp}{44513596sp}{square}
\tracepos{0/58:case:code[2]:19,2}{30229437sp}{43808176sp}{square}
\boxchar{0/58:case:code[2]:19,2}{1}
\tracepos{0/58:case:code[2]:19,3}{29524017sp}{43399484sp}{square}
\tracepos{0/58:case:code[2]:19,3}{30229437sp}{42694064sp}{square}
\boxchar{0/58:case:code[2]:19,3}{2}
\tracepos{0/58:case:code[2]:19,4}{29524017sp}{42285372sp}{square}
\tracepos{0/58:case:code[2]:19,4}{30229437sp}{41579952sp}{square}
\boxchar{0/58:case:code[2]:19,4}{3}
\tracepos{0/58:case:code[2]:19,5}{29524017sp}{41171260sp}{square}
\tracepos{0/58:case:code[2]:19,5}{30229437sp}{40465840sp}{square}
\boxchar{0/58:case:code[2]:19,5}{4}
\tracepos{0/58:case:code[2]:19,6}{29524017sp}{40057148sp}{square}
\tracepos{0/58:case:code[2]:19,6}{30229437sp}{39351728sp}{square}
\boxchar{0/58:case:code[2]:19,6}{5}
\tracepos{0/58:case:code[1]:20,1}{31015874sp}{45627708sp}{square}
\tracepos{0/58:case:code[1]:20,1}{31721294sp}{44922288sp}{square}
\boxchar{0/58:case:code[1]:20,1}{0}
\tracepos{0/58:case:code[1]:20,2}{31015874sp}{44513596sp}{square}
\tracepos{0/58:case:code[1]:20,2}{31721294sp}{43808176sp}{square}
\boxchar{0/58:case:code[1]:20,2}{1}
\tracepos{0/58:case:code[1]:20,3}{31015874sp}{43399484sp}{square}
\tracepos{0/58:case:code[1]:20,3}{31721294sp}{42694064sp}{square}
\boxchar{0/58:case:code[1]:20,3}{2}
\tracepos{0/58:case:code[1]:20,4}{31015874sp}{42285372sp}{square}
\tracepos{0/58:case:code[1]:20,4}{31721294sp}{41579952sp}{square}
\boxchar{0/58:case:code[1]:20,4}{3}
\tracepos{0/58:case:code[1]:20,5}{31015874sp}{41171260sp}{square}
\tracepos{0/58:case:code[1]:20,5}{31721294sp}{40465840sp}{square}
\boxchar{0/58:case:code[1]:20,5}{4}
\tracepos{0/58:case:code[1]:20,6}{31015874sp}{40057148sp}{square}
\tracepos{0/58:case:code[1]:20,6}{31721294sp}{39351728sp}{square}
\boxchar{0/58:case:code[1]:20,6}{5}

```

type	English	French
command	<code>\namefield</code>	<code>\champnom</code>
environment	<code>choices</code>	<code>reponses</code>
environment	<code>choiceshoriz</code>	<code>reponseshoriz</code>
environment	<code>choicescustom</code>	<code>reponsesperso</code>
command	<code>\correctchoice</code>	<code>\bonne</code>
command	<code>\wrongchoice</code>	<code>\mauvaise</code>
command	<code>\lastchoices</code>	<code>\alafin</code>
command	<code>\AMCIntervals</code>	<code>\choixIntervalles</code>
command	<code>\scoring</code>	<code>\bareme</code>
command	<code>\scoringDefaultM</code>	<code>\baremeDefautM</code>
command	<code>\scoringDefaultS</code>	<code>\baremeDefautS</code>
command	<code>\onecopy</code>	<code>\exemplaire</code>
environment	<code>examcopy</code>	<code>copieexamen</code>
command	<code>\shufflegroup</code>	<code>\melangegroupe</code>
command	<code>\insertgroup</code>	<code>\restituegroupe</code>
command	<code>\AMCform</code>	<code>\formulaire</code>
command	<code>\AMCformBegin</code>	<code>\AMCdebutFormulaire</code>
option	<code>noshuffle</code>	<code>ordre</code>
option	<code>answers</code>	<code>correc</code>
option	<code>indivanswers</code>	<code>correcindiv</code>
option	<code>box</code>	<code>bloc</code>
option	<code>separateanswersheet</code>	<code>ensemble</code>
option	<code>digits</code>	<code>chiffres</code>

Table 1: French equivalent commands

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Samples</b>	<b>1</b>
2.1	Standard layout . . . . .	4
2.2	Separate answer sheet . . . . .	5
2.3	Without markers . . . . .	6
<b>3</b>	<b>Usage</b>	<b>10</b>
3.1	Package options . . . . .	10
3.2	Questions and answers . . . . .	11
3.3	Scoring . . . . .	13
3.4	Groups of questions . . . . .	14
3.5	Students identification . . . . .	15
3.6	Separate answer sheet . . . . .	16
3.7	Random computation questions . . . . .	17
3.8	French command names . . . . .	20
3.9	Customisation . . . . .	21
3.9.1	Boxes . . . . .	21
3.9.2	Codes . . . . .	21
3.9.3	Answers . . . . .	22
<b>4</b>	<b>Implementation</b>	<b>22</b>
4.1	Variables . . . . .	23
4.2	Dimensions . . . . .	25
4.3	Human readable sheet ID position . . . . .	26
4.4	Localisation . . . . .	26
4.4.1	English . . . . .	26
4.4.2	Catalan . . . . .	27
4.4.3	Dutch . . . . .	27
4.4.4	French . . . . .	28
4.4.5	German . . . . .	28
4.4.6	Italian . . . . .	28
4.4.7	Norwegian . . . . .	29
4.4.8	Portuguese . . . . .	29
4.4.9	Spanish . . . . .	29
4.4.10	Japanese . . . . .	30
4.4.11	Other languages . . . . .	30
4.5	Interaction with other packages . . . . .	30
4.5.1	cleveref . . . . .	30
4.6	Random . . . . .	30
4.6.1	Random pseudo-generator . . . . .	30
4.6.2	Uniform random deviates . . . . .	31
4.6.3	Tokens shuffling . . . . .	31
4.7	Keys numbering . . . . .	32
4.8	Boxes . . . . .	32
4.8.1	Character logging . . . . .	32
4.8.2	Position logging . . . . .	32
4.8.3	Boxes to be checked by students . . . . .	34

4.8.4	Scoring zones	40
4.8.5	Binary boxes	40
4.9	Checking Environment	42
4.10	Handling groups of questions	42
4.11	Questions	46
4.11.1	Managing answers	47
4.11.2	Separate answer sheet	47
4.11.3	Formatting answers	51
4.11.4	Score zones	52
4.11.5	Formatting questions	55
4.11.6	Explanations	57
4.12	Scoring	57
4.13	Numerical data	58
4.13.1	Codes	58
4.13.2	Numerical questions	61
4.13.3	Intervals	75
4.14	Open questions	76
4.15	Boxes with letters only	79
4.16	Page formatting	80
4.16.1	Watermark	80
4.16.2	Signs for scan analysis	80
4.17	Defining a single exam copy content	82
4.18	Pre-association	83
4.19	Package options	84
4.20	Survey add-on	85
4.21	Package Errors	87
4.22	Optional features	87
4.23	Use with recent LuaTeX versions	87
4.24	External control	88
4.25	Page layout	88
4.26	Initialisation	89
4.27	French command names	89
<b>5</b>	<b>Outputs</b>	<b>90</b>
5.1	namefield command	90
5.2	AMCboxedchar command	90
5.3	AMCcode command	90

# Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

<b>Symbols</b>			1399,	1400,	1420,	1824,	1831,	1841,
\"	.....	148, 154	1421,	1426,	1427,	1842,	1843,	1844,
\@aucune	.....	839, 843, 844	1428,	1429,	1430,	1845,	1846,	1847,
\@firstoftwo	.....	659	1431,	1436,	1438,	1848,	1849,	1850,
\@ifstar	.....	1003, 1004	1439,	1440,	1441,	1852,	1857,	1858,
\@secondoftwo	.....	661	1442,	1444,	1446,	1861,	1867,	1869,
\@skiphyperreffalse	...	1172	1447,	1448,	1449,	1871,	1872,	1875,
\@skiphyperreftrue		1168, 1172	1452,	1459,	1460,	1876,	1880,	1881,
\@tempboxa	.	2237, 2241, 2242	1461,	1463,	1465,	1882,	1884,	1885,
\@tempdimb	.....	2226, 2228	1466,	1468,	1470,	1887,	1888,	1889,
\@tempdimc	.....	2227, 2228	1472,	1475,	1476,	1890,	1891,	1893,
\_	.....	1207	1478,	1480,	1481,	1894,	1897,	1899,
\~	.....	189, 190, 195, 196	1482,	1484,	1490,	1904,	1905,	1906,
<b>A</b>			1495,	1507,	1541,	1907,	1909,	1910,
\aa	.....	175	1543,	1545,	1547,	1911,	1912,	1913,
\alafin	.....	21, 2636	1549,	1552,	1555,	1914,	1915,	1918,
\amc	.....	866, 877,	1558,	1561,	1564,	1919,	1920,	1921,
		879, 881, 882, 884,	1567,	1570,	1573,	1925,	1927,	1929,
		887, 888, 890, 893,	1576,	1578,	1580,	1932,	1937,	1938,
		894, 897, 898, 899,	1583,	1592,	1594,	1939,	1940,	1941,
		900, 902, 904, 905,	1602,	1606,	1607,	1942,	1943,	1945,
		907, 909, 910, 912,	1608,	1614,	1616,	1946,	1947,	1948,
		914, 915, 918, 919,	1628,	1629,	1630,	1952,	1953,	1955,
		921, 922, 924, 925,	1631,	1632,	1633,	1961,	1968,	1973,
		927, 928, 930, 931,	1634,	1635,	1638,	1975,	1976,	1982,
		933, 936, 937, 939,	1641,	1645,	1648,	1991,	1993,	1994,
		941, 945, 946, 947,	1651,	1655,	1658,	1995,	1996,	2000,
		948, 1272, 1273, 1274,	1662,	1665,	1670,	2001,	2002,	2006,
		1275, 1276, 1277,	1672,	1674,	1681,	2007,	2008,	2009,
		1278, 1280, 1287,	1683,	1686,	1695,	2011,	2012,	2013,
		1290, 1295, 1299,	1696,	1697,	1698,	2026,	2028,	2034,
		1307, 1310, 1311,	1700,	1702,	1705,	2043,	2046,	2047,
		1312, 1315, 1320,	1710,	1711,	1713,	2053,	2054,	2055,
		1322, 1325, 1326,	1717,	1719,	1722,	2056,	2057,	2058,
		1327, 1328, 1330,	1723,	1724,	1725,	2059,	2061,	2062, 2065
		1332, 1334, 1335,	1727,	1728,	1729,	\AMC@addpages	2349, 2351, 2353	
		1337, 1338, 1342,	1732,	1735,	1740,	\AMC@affecte	.....	<u>279</u> , 1183
		1343, 1344, 1346,	1742,	1745,	1749,	\AMC@amclog	.....	<u>17</u>
		1347, 1348, 1350,	1750,	1751,	1753,	\AMC@answerBox	.....	<u>478</u>
		1355, 1357, 1359,	1756,	1761,	1767,	\AMC@answerBox@	...	<u>359</u> ,
		1367, 1372, 1373,	1769,	1771,	1774,		486, 533, 535, 551,	
		1374, 1376, 1378,	1782,	1783,	1790,		553, 619, 621, 629, 631	
		1381, 1382, 1383,	1791,	1793,	1794,	\AMC@binaryBoxes	.....	652
		1384, 1386, 1388,	1795,	1797,	1800,	\AMC@binaryCode	...	<u>602</u> ,
		1391, 1392, 1394,	1801,	1804,	1805,		653, 2257, 2258, 2259	
		1395, 1397, 1398,	1807,	1813,	1817,	\AMC@box	.....	<u>520</u> , 1053,
			1818,	1820,	1823,		1054, 1057, 2493, 2494	



\AMC@boxcolor	377, 503, 2148	\AMC@lang@code	2384, 2419, 2420, 2616	\AMC@mem@add	907, 953, 960, 967, 974, 1406
\AMC@boxcolor@	377, 378, 379, 381, 399, 408, 420, 424, 441	\AMC@loc@CA	105	\AMC@mem@add@ifneeded	864, 1214
\AMC@boxeddown	391, 491, 502	\AMC@loc@catalog	99, 112, 126, 139, 153, 167, 180, 192, 206, 218, 2414	\AMC@mem@addsingle@ifneeded	956, 996, 998, 1000, 1002
\AMC@boxedheight	404, 418, 419, 421, 422, 425, 426, 470, 493, 497, 498, 510, 513	\AMC@loc@corrected	98, 111, 125, 138, 152, 166, 179, 191, 205, 217, 2253	\AMC@mem@addvar	912
\AMC@boxedrule	396, 418, 419, 420, 489, 500	\AMC@loc@DE	146	\AMC@mem@aid	931, 966
\AMC@boxedwidth	407, 418, 419, 421, 422, 425, 426, 470, 492, 497, 499, 510, 511, 1718	\AMC@loc@draft	93, 106, 120, 133, 147, 161, 174, 187, 200, 213, 2224	\AMC@mem@answer	963, 1026, 1033, 1034, 1035, 1040, 1046
\AMC@catalogMode	2409, 2417, 2576	\AMC@loc@ES	199	\AMC@mem@category	925, 959, 2038, 2184, 2220
\AMC@CBtaille	600, 2281, 2423	\AMC@loc@explain	100, 113, 140, 154, 193, 219, 1246	\AMC@mem@clear	882, 2341
\AMC@checkbox	359, 380, 383, 433, 435, 445, 451, 461, 619, 621, 1053, 1619, 1639, 1651, 1655, 2493	\AMC@loc@FR	132, 2419, 2631	\AMC@mem@next	902, 958, 972
\AMC@chiffres	1260	\AMC@loc@IT	160	\AMC@mem@openQuestion	970, 1213
\AMC@crosschar	402, 505	\AMC@loc@JA	212	\AMC@mem@qidaffname	919, 973
\AMC@crossrule	424, 490, 506	\AMC@loc@message	94, 107, 121, 134, 148, 162, 175, 188, 201, 214, 2594	\AMC@mem@show	946, 983, 992
\AMC@definitnumero	280, 284	\AMC@loc@namesurname	104, 117, 130, 144, 158, 171, 184, 197, 210	\AMC@mem@show@filter	947, 987
\AMC@draw@crossfalse	382	\AMC@loc@NL	119	\AMC@mn@leftmargin	1063, 1071, 1074, 1080
\AMC@draw@crosstrue	384	\AMC@loc@NO	173	\AMC@mn@rightmargin	1064, 1072, 1075, 1078
\AMC@error@explain	1246, 1248, 2519	\AMC@loc@none	101, 114, 127, 141, 155, 168, 181, 194, 207, 220, 839	\AMC@mn@sep	1062, 1078, 1080
\AMC@fillcolor@	380, 398, 399, 415, 420	\AMC@loc@PT	186	\AMC@mn@test	1061, 1068, 1077
\AMC@fin@rep	826, 1011, 1015, 1019, 1022, 1025	\AMC@loc@q	97, 110, 124, 137, 151, 165, 178, 190, 204, 216, 1175	\AMC@multi@report	1325, 2347
\AMC@formBox	520	\AMC@loc@qf	96, 109, 123, 136, 150, 164, 177, 189, 203, 215, 848	\AMC@multiclear	1330, 2338
\AMC@formBox@	520	\AMC@loc@question	102, 115, 128, 142, 156, 169, 182, 195, 208, 221, 227	\AMC@NCBcheck	596, 2259, 2423
\AMC@fullGroupsfalse	34	\AMC@loc@questions	103, 116, 129, 143, 157, 170, 183, 196, 209, 222, 227	\AMC@NCBetud	596, 2257, 2422
\AMC@fullGroupstrue	2387, 2413	\AMC@logchar	287, 388	\AMC@NCBpage	596, 2258, 2422
\AMC@if@separate@question	877, 952, 957, 965, 971, 1405, 2038, 2184, 2220	\AMC@logfile	17, 18, 19, 288	\AMC@new@savebox	360, 2448, 2452, 2453, 2454, 2455
\AMC@imax	713, 725, 726, 727	\AMC@LR	21, 390, 2229, 2276, 2277, 2278, 2279, 2280, 2594	\AMC@note	2254, 2255, 2289, 2299, 2593
\AMC@intituleHead	2253, 2270, 2414	\AMC@makeovalbox	413, 432, 433, 434, 435	\AMC@numeric@alsocorrect	1598, 1964, 1968, 1978, 1982
\AMC@keepmemoryfalse	2339			\AMC@numeric@keepas	1596, 1957, 1958
\AMC@keepmemorytrue	993			\AMC@numeric@scoreapprox	1588, 1923, 1933
\AMC@keyBox@	549, 1298, 1621, 1624			\AMC@numeric@scoreexact	1586, 1922, 1928
\AMC@keys@line	2416			\AMC@numeric@scorewrong	1590, 1930, 1933
\AMC@keys@next	2415			\AMC@numeration	279, 283, 284
				\AMC@outside@sep	495, 501
				\AMC@oval@radius	420, 494, 511, 513
				\AMC@ovalbox@	434, 448, 2454

\AMC@ovalbox@F	435, 446, 2455	\AMC@SR@time	246, 2400	\AMC@sza@height	1119, 1138, 1143
\AMC@ovalbox@R	431, 432, 454, 2452	\AMC@SR@advance	232, 239, 240	\AMC@sza@init@margin	1129
\AMC@ovalbox@RF	433, 452, 2453	\AMC@SR@bit	239	\AMC@sza@init@margin	1133
\AMC@pagepos	293, 2266	\AMC@SR@const	230, 234, 237	\AMC@sza@init@none	1121
\AMC@premierecopie	601, 2323, 2326, 2424	\AMC@SR@max	245, 273	\AMC@sza@init@question	1125
\AMC@prepare	282, 285, 286	\AMC@SR@nextByte	245	\AMC@sza@width	1119, 1137, 1142
\AMC@prepare@element	676, 684, 799	\AMC@SR@num	247, 248, 250, 251, 255, 258, 273, 274	\AMC@tempenv	657, 658
\AMC@printformoutside@false	524	\AMC@SR@set	231, 243, 244, 246	\AMC@tracebox	293, 347, 440, 466, 584, 586, 589, 1042, 2247
\AMC@printformoutside@true	526, 527	\AMC@SR@test	240, 251	\AMC@tracechar	335, 537, 555
\AMC@printkeyoutside@false	541	\AMC@SR@value	242	\AMC@tracepos	294, 405, 411, 1028, 1031, 1205, 1214
\AMC@printkeyoutside@true	543, 545	\AMC@stepQuestion	1165, 1199, 1239	\AMC@tracexosx	303, 323
\AMC@qaff	849, 1165, 1207, 1208, 1242	\AMC@sti	260, 268, 272, 275, 276	\AMC@tracexosy	312, 322, 324
\AMC@save@box	361, 416, 2449	\AMC@stil	261, 269, 270, 271, 273, 277	\AMC@unnumero	279
\AMC@setcolors@	376, 397, 414, 439	\AMC@surveyfalse	2433	\AMC@use@box	362, 446, 448, 452, 454, 2450
\AMC@shape@form	473	\AMC@surveytrue	2434	\AMC@VERSION	2608, 2610
\AMC@shape@form@base	460, 474	\AMC@sz@box	1084, 1086, 1088, 1091, 1134	\AMC@watertext	2224, 2235
\AMC@shape@form@ticked	462, 464, 467	\AMC@sz@callin	1096, 1101, 1109, 1111, 1176, 1209, 1210	\AMC@XYFILE	296, 305, 314, 326, 331, 332, 333, 337, 343, 345, 1186, 1208, 1322, 2331, 2375, 2380, 2381, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2618, 2620
\AMC@shape@none	477	\AMC@sz@callin@question	1086	\AMC@addpagesto	2349
\AMC@shape@oval	438	\AMC@sz@callout	1095, 1100, 1104, 1106, 1202, 1203	\AMC@answer	1046, 1047, 2137, 2199
\AMC@shape@square	395	\AMC@sz@callout@margin	1088	\AMC@association	2373
\AMC@shapename	293, 301, 310, 319, 496	\AMC@sz@callout@margin	1091	\AMC@beforeQuestion	1174, 1201, 1282
\AMC@shapename@	293, 391, 507	\AMC@sz@depth	1084, 1094, 1099	\AMC@beginAnswer	1018, 1047, 2136, 2198
\AMC@shapeprepare	507, 515, 2456	\AMC@sz@height	1084, 1093, 1098	\AMC@beginQuestion	1174, 1207, 1222, 1228, 1242, 1281
\AMC@shapeprepare@form	459	\AMC@sz@init@margin	1090	\AMC@bin@begin	651
\AMC@shapeprepare@none	476	\AMC@sz@width	1084, 1092, 1097	\AMC@bin@did	606, 614, 617, 619, 624, 627, 629, 635
\AMC@shapeprepare@oval	430	\AMC@sza@box	1119, 1127, 1130	\AMC@bin@digit	604, 637, 640, 646, 649
\AMC@shapeprepare@square	394	\AMC@sza@callin	858, 859, 1141, 1146, 1154, 1156	\AMC@bin@hsep	608, 611, 615, 625
\AMC@shuffletoks	260, 746, 749, 819	\AMC@sza@callin@margin	1131	\AMC@bin@id	605, 610, 618, 619, 628, 629
\AMC@smashbox	364, 367, 368, 369, 370, 371, 374	\AMC@sza@callin@margin	1135	\AMC@bin@ndigits	607, 609, 646, 649
\AMC@smashboxheight	365, 368, 369, 370, 373	\AMC@sza@callin@none	1123		
\AMC@smashcentered	366, 408, 409, 441, 442	\AMC@sza@callin@question	1127		
\AMC@SR	229, 231, 234, 235, 236, 237, 239, 241, 242, 2332	\AMC@sza@callout	854, 855, 1140, 1145, 1149, 1151		
\AMC@SR@count	234, 235, 237, 245, 249, 252, 255, 256, 257, 258	\AMC@sza@callout@margin	1130		
		\AMC@sza@callout@margin	1134		
		\AMC@sza@callout@none	1122		
		\AMC@sza@callout@question	1126		
		\AMC@sza@depth	1119, 1139, 1144		

\AMCbin@number . . . . .	\AMCendAnswer . . . . .	\AMCid@name . 65, 565, 568,
. 603, 636, 639, 641, 643	. 1019, 1047, 2136, 2198	570, 574, 576, 916,
\AMCbin@one . . . . . 613, 641	\AMCexponent 1531, 1998, 2010	973, 1028, 1031, 1042,
\AMCbin@sequence . . 602,	\AMCexternalQuestion . . 1189	1183, 1186, 1214,
636, 641, 642, 647, 648	\AMCform . . . . . 16, 864, 2637	1298, 1621, 1624, 2080
\AMCbin@style . . . . . 612, 636	\AMCformAfterQuestion . .	\AMCid@quest . . . . . 26,
\AMCbin@zero . . 623, 642, 647	. . . . . 847, 1214	331, 332, 333, 565,
\AMCbloc . . . . . 1181	\AMCformAnswer . . . . . 846	568, 570, 574, 576,
\AMCbo@help . 2188, 2189, 2202	\AMCformAnswerA . . . 863, 967	584, 586, 589, 915,
\AMCbo@hide . . . . . 2215, 2221	\AMCformatChoices . . . . .	973, 1028, 1031,
\AMCbo@opts . . . . . 2192	. 1403, 2039, 2185, 2221	1042, 1183, 1185,
\AMCbo@show . . . . . 2193, 2221	\AMCformBeforeQuestion .	1186, 1205, 1208,
\AMCbotextGoto . . 2187, 2216	. . . . . 846, 852	1214, 1298, 1622,
\AMCboxColor . . . . . 518, 2281	\AMCformBegin . 16, 864, 2638	1625, 1854, 2119, 2195
\AMCboxDimensions . 519, 636	\AMCformFilter . . . . . 985	\AMCIDBoxesA . . . . . 2257, 2261
\AMCBoxedAnswers 1005, 2391	\AMCformHSpace 66, 862, 2199	\AMCIDBoxesABC . . 2260, 2283
\AMCBoxOnly . . . . . 2187	\AMCformQuestion . . . . . 846	\AMCIDBoxesB . . . . . 2258, 2262
\AMCboxOutsideLetter . .	\AMCformQuestionA . 850, 974	\AMCIDBoxesC . . . . . 2259, 2262
. . . . . 520, 2138, 2200	\AMCformQuestionN . 849, 857	\AMCids@height . 77, 88, 2282
\AMCboxStyle . . . . .	\AMCformS . . . . . 864	\AMCids@sidefalse . . . 80, 82
. . . . . 21, 489, 2391, 2441	\AMCformVSpace . . . . . 66, 846	\AMCids@sidetrue . . . . . 84
\AMCccircle . . . . . 2243, 2246	\AMCgroup@pre . . . . . 742, 765	\AMCids@topfalse . . . . 80, 84
\AMCchoiceLabel . . . 478, 559	\AMCgrouploop@next . . . . .	\AMCids@toptrue . . . . . 82
\AMCchoiceLabelFormat . .	. . . . . 767, 780, 798	\AMCids@width . . 76, 87, 2285
. . . . . 392, 478	\AMCgrouploop@prep . . . . .	\AMCidsPosition . . . . . 74
\AMCcleardoublepage 16, 2358	. . . . . 753, 778, 796	\AMCidsVar . . . . . 78
\AMCcode . . . . . 1399	\AMCgrouppre@cyclic . . . 736	\AMCidsVarN . . . . . 78, 79
\AMCcodeGrid . . . . . 15, 1260	\AMCgrouppre@fixed . . . . 698	\AMCif@env . . 656, 1246, 1248
\AMCcodeGridInt . . . 15, 1260	\AMCgrouppre@withoutreplacement . . . . . 714	\AMCifcategory . . . . . 948
\AMCcodeH . . . . . 1400	\AMCgrouppre@withreplacement . . . . . 705	\AMCinterBquest . . . . 71, 1214
\AMCcodeHspace . . . . .	. . . . . 1036, 1037, 1045	\AMCinterBrep . . . . . 66, 1032
. . . . . 1261, 1302, 1361	\AMChorizAnswerSep . . . . .	\AMCinterIquest . . . . 70, 1214
\AMCcodeID . . . . . 1266, 1341	. . . . . 1038, 1039, 1042, 1044	\AMCinterIrep . . . . . 66, 1009
\AMCcodeID@@ . . . . .	\AMCid@check . . . . .	\AMCIntervalFormat 2050, 2061
. 1265, 1269, 2432, 2582	. 27, 328, 2259, 2264,	\AMCIntervals 17, 2050, 2640
\AMCcodeID@dot . . . . . 1264	2265, 2282, 2286, 2600	\AMClabel . . . . . 2366, 2371
\AMCcodeID@mode . 1267, 2609	\AMCid@checkmax . . . . .	\AMCload@@reponse . 830,
\AMCcodeID@squarebrackets . . . . . 1263	. . 596, 2265, 2422, 2600	832, 1053, 1057, 2493
\AMCcodeVspace . . . . .	\AMCid@etud . 28, 290, 298,	\AMCload@counter . . . . .
. 1262, 1300, 1360, 1363	307, 316, 328, 331,	. . 25, 808, 809, 810,
\AMCcompleteMulti . . 12, 63	332, 333, 338, 353,	811, 814, 815, 819,
\AMCcurrentenv 655, 658, 1198	474, 1208, 2256, 2257,	822, 823, 824, 825, 828
\AMCdebutFormulaire 21, 2638	2282, 2286, 2294,	\AMCload@reponse . . 813, 832
\AMCdecimalPoint . . . . .	2322, 2323, 2325,	\AMClocalized . . . . . 92
. . . . . 1529, 1714, 1746	2332, 2337, 2344,	\AMCloop@k . . . . 719, 725,
\AMCdefault@groupmode . .	2345, 2365, 2366, 2375	729, 731, 752, 760, 769
. . . . . 673, 687, 688	\AMCid@etudfin . . . . . 30,	\AMCmarginNote . . . . .
\AMCdontAnnotate . 331, 2153	2324, 2325, 2326, 2344	. . . . . 1065, 1091, 1134
\AMCdontScan . . . . . 331, 2151	\AMCid@etudstart . . . . .	\AMCmem@elt@cat 922, 934, 938
\AMCdump@reponses . 819, 833	. . . . . 29, 2322, 2345	\AMCmessage . . . . . 17, 280,
\AMCemptybox 1059, 1084, 1119		592, 992, 1051, 1056,
		1185, 1192, 1193,
		1194, 1214, 1218,

1223, 1229, 1256,	\AMCocol@Background . . . .	816, 836, 838, 842,
1257, 1258, 1259,	. . . . . 2070, 2077, 2142	928, 966, 1007, 1013,
1297, 1411, 1618,	\AMCocol@BoxFrameRule . .	1017, 1021, 1024,
1650, 1652, 1657,	. . . . . 2071, 2095, 2142	1028, 1031, 1042,
1659, 1664, 1666,	\AMCocol@Foreground . . . .	1050, 1051, 1054,
1675, 1688, 1704,	. . . . . 2073, 2078,	1055, 1056, 1057,
1796, 1799, 1832,	2147, 2148, 2158, 2161	1292, 1296, 1297,
1834, 1854, 1919,	\AMCocol@FrameRule . . . . .	1298, 1404, 1406,
1926, 1943, 1945,	. . . . . 2072, 2096, 2169	1617, 1618, 1622,
1948, 1950, 1953,	\AMCOpen . . . . . 2068	1625, 2490, 2491, 2494
1955, 1958, 1962,	\AMCopen@answer . . . . .	\AMCrep@fini 830, 833, 837, 845
1965, 1967, 1974,	. 2074, 2081, 2110, 2126	\AMCrep@init . . 826, 1010,
1979, 1981, 2021,	\AMCopen@boxframerule . .	1013, 1017, 1021, 1024
2119, 2176, 2195,	. . . . . 2093, 2094, 2141	\AMCrep@itemize . 1007, 1026
2212, 2320, 2337,	\AMCopen@boxmargin . . . . .	\AMCrep@ligne . . 1013, 1026
2440, 2444, 2491,	. . . . . 2091, 2092, 2140	\AMCrep@o . . . . . 827, 829
2601, 2602, 2603, 2604	\AMCopen@contentcommand	\AMCrep@perso . . . 1017, 1026
\AMCncol@Background . . . .	. . . . . 2082, 2170	\AMCrep@r . . . . . 831
. . . . . 1539, 1990	\AMCopen@framerule . . . . .	\AMCrep@tikz . . . . 1021, 1026
\AMCncol@Border . 1535, 1990	. . . . . 2097, 2098, 2168	\AMCrep@tikzmat . 1024, 1026
\AMCncol@BorderWidth . . .	\AMCopen@Hspace . . 2083,	\AMCreTick . . . . . 331, 2152
. . . . . 1537, 1989	2084, 2139, 2144, 2156	\AMCrien@deux . . . . 813, 830
\AMCneeds@storeboxfalse 2429	\AMCopen@LineHeight . . . .	\AMCscoreZone . . . . 580, 1059
\AMCneeds@storeboxtrue 2430	. . . . . 2087, 2088, 2109	\AMCscoreZoneAnswerSheet
\AMCnobloc . . . . . 1180, 1286	\AMCopen@Lines . . . . .	. . . . . 1059
\AMCnoCompleteMulti . . . .	. . . . . 2089, 2090, 2114	\AMCsection . . . . . 995
. . . . . 12, 64, 2135	\AMCopen@lines . . . . . 2107	\AMCsectionNumbered 995, 1003
\AMCNombreCopies . . . . .	\AMCopen@lineuptext 2076,	\AMCsectionStar . . 999, 1003
. 2318, 2319, 2617, 2620	2099, 2124, 2127, 2130	\AMCsetFoot . . . . . 2255
\AMCnoScoreZone . 1164, 1283	\AMCopen@question . . . . .	\AMCsetScoreZone . . . . .
\AMCnontextGoto . . 1520, 2030	. 2075, 2080, 2143, 2144	. . . . . 1117, 1118, 1164
\AMCnontextSign 1527, 1765, 1766	\AMCopen@Width . . . . .	\AMCsetScoreZoneAnswerSheet
\AMCnontextVHead . . . . .	. . . . . 2085, 2086, 2108	. . . . . 1162, 1163, 1164
. . . . . 1521, 1718, 1864	\AMCopenHide . . . . 2179, 2185	\AMCshowSignificantDigits
\AMCnum@copies . 31, 2318,	\AMCopenOpts . . . . . 2106	. . . . . 1468
2319, 2320, 2321, 2324	\AMCopenShow . . . . 2117, 2185	\AMCsignificantDigits . 1463
\AMCnum@Questions . 820, 822	\AMCotextGoto . . 2068, 2180	\AMCstartWithQuestion . .
\AMCnumeric@Hspace . . . . .	\AMCotextReserved 2069,	. . . . . 1182, 1190, 1200
. . 1522, 1534, 1676,	2079, 2155, 2159, 2161	\AMCstudentlabel . . . . .
1678, 1714, 1733, 2010	\AMCoutsideLabelFormat 520	. 2366, 2367, 2368, 2369
\AMCnumeric@Vspace . . . . .	\AMCpageref . . . . . 2369	\AMCStudentNumber . . . . 2256
. . . . . 1523, 1533,	\AMCpostNquest . . . . 72, 2019	\AMCstudentslistfile . . 2378
1649, 1656, 1663,	\AMCpostOquest . . . . 73, 2172	\AMCsubjectPageTag 2293, 2298
1690, 1706, 1720,	\AMCqlabel . . . . . 2370	\AMCsubsection . . . . . 995
1754, 1766, 1997, 1999	\AMCquestionaff . . . . . 1165	\AMCsubsectionNumbered .
\AMCnumericChoices . 18, 1403	\AMCquestionNumberfalse 1284	. . . . . 997, 1004
\AMCnumericChoicesPlain	\AMCquestionNumbertrue . 45	\AMCsubsectionStar 1001, 1004
. . . . . 2037, 2044	\AMCrandomseed . . . . . 243	\AMCsw@p . . . . . 260
\AMCnumericHide . 2034, 2039	\AMCref . . . . . 2366, 2366	\AMCsw@p@ . . . . 262, 264, 266
\AMCnumericOpts . . . . . 1614	\AMCrep@@count 834, 836, 838	\AMCsz@loggedfalse . . . . 580
\AMCnumericShow . 2026, 2039	\AMCrep@bloc . . . . 1008, 1026	\AMCsz@loggedtrue . . . . 593
\AMCnumero . . . . . 1166, 2336	\AMCrep@count . . 565, 568,	\AMCtext . . . . . 91
	570, 574, 576, 807,	

\AMCtok@ik .....		\choicescustom .....	2625	2047, 2055, 2065,
. 719, 720, 722, 727,		choicescustom (env.)	12, 1005	2540, 2541, 2543,
728, 729, 730, 731,		\choiceshoriz .....	2624	2544, 2546, 2547,
751, 759, 772, 781, 800		choiceshoriz (env.)	12, 1005	2550, 2551, 2553, 2554
\AMCtok@k .....	664,	\choixIntervalles	21, 2640	\csvreader .....
679, 680, 681, 685, 800		\cleargroup .....	14, 788	. 2465, 2470, 2476, 2485
\AMCtok@max .....	665	\clist .....	1272,	
\AMCtok@size .....		1278, 1287, 1322,		<b>D</b>
. 666, 754, 755, 756,		1335, 1339, 1373,		\define@boolkey .....
757, 765, 773, 782, 801		1382, 1383, 1477,		... 504, 2100, 2101,
\AMCw@termark ...	2223, 2267	1480, 1491, 1492,		2102, 2103, 2104, 2190
\AMCw@terprint ..	2223, 2267	1576, 1577, 1711,		\define@choicekey .....
amcxyfile (env.)	342	1712, 1742, 1743,		... 78, 496, 1102, 1147
\AMCzone .....	347, 348	1843, 1844, 1861, 1869		digits (option) .....
\answer .....	289, 1050, 2458	\codeDigitExterne	2581, 2582	10
answers (option) .....	10	completemulti (option)	.. 10	\ding .....
asbox (option) .....	10	\coordinate .....	1067	468
\association .....	2375	\copieexamen .....	2633	\dontannotate .....
auto (env.) .....	2458	copieexamen (env.)	21	332
automarks (option) .....	11	\copygroup .....	14, 788	\dontscan .....
		\copygroupfrom .....	14, 788	331
<b>B</b>		correc (option) .....	21	\dotfill ....
\bareme .....	21, 2628	correcindiv (option)	21	356, 2112, 2128
\baremeDefaultM ....	21, 2629	\correctchoice .....		\draw ...
\baremeDefaultS ....	21, 2630	13, 844, 1050, 2060, 2626		420, 424, 1078, 1080
\bf .....	1530	\CorrigeExterne .....	2558	
bloc (option) .....	21	\CorrigeIndivExterne ..	2558	<b>E</b>
\bonne .....	21, 2626	\crefalias .....	226	\element .....
\bool 940, 1275, 1276, 1277,		\crefname .....	227	14, 664
1299, 1310, 1311,		\cs 877, 881, 882, 884, 887,		\endchoices .....
1334, 1337, 1341,		890, 893, 897, 902,		\endchoicescustom ....
1348, 1350, 1359,		904, 907, 909, 912,		2625
1607, 1672, 1683,		914, 918, 919, 921,		\endchoiceshoriz ....
1698, 1717, 1722,		924, 925, 927, 930,		2624
1763, 1767, 1782,		931, 933, 936, 945,		\endcopieexamen .....
1795, 1801, 1805,		946, 947, 948, 1280,		2633
1820, 1830, 1831,		1290, 1295, 1320,		\endexamcopy .....
1871, 1875, 1884,		1325, 1327, 1330,		2633
1925, 1961, 1973, 1991		1332, 1372, 1376,		\endreponses .....
box (option) .....	10	1381, 1386, 1391,		2623
\boxchar .....	338	1394, 1397, 1398,		\endreponseshoriz ....
\boxput .....	6, 401, 440	1399, 1400, 1418,		2624
		1426, 1436, 1439,		\endreponsesperso ....
		1444, 1448, 1460,		2625
		1463, 1465, 1468,		ensemble (option) .....
		1470, 1476, 1490,		21
		1495, 1507, 1602,		environments:
		1606, 1614, 1616,		amcxyfile .....
		1630, 1645, 1670,		342
		1681, 1696, 1710,		auto .....
		1740, 1761, 1790,		2458
		1852, 2026, 2028,		choices .....
		2034, 2043, 2046,		12, 1005
				choicescustom ..
				12, 1005
				choiceshoriz ...
				12, 1005
				copieexamen .....
				21
				examcopy .....
				2530
				question .....
				11, 1179
				question-auto ....
				2458
				questionmult ...
				11, 1179
				Questionnaires ....
				2458
				questionouverte ...
				1179
				reponses .....
				21
				reponseshoriz .....
				21
				reponsesperso ....
				21
				tikz-multi .....
				1005
				tikz-single .....
				1005
				values .....
				2458
				values-auto .....
				2458
				variable-auto ....
				2458
				variable-multi ...
				1179
				variable-single ...
				1179



<code>\evensidemargin</code> . 1072, 1074	333, 336, 1051, 1056,	<code>\ifAMC@qbloc</code> . . . 41, 1174,
<code>\examcopy</code> . . . . . 2633	1184, 1191, 1208,	1205, 1214, 1241,
<code>examcopy (env.)</code> . . . . . 2530	1217, 1256, 1257,	1243, 2031, 2181, 2217
<code>\exemplaire</code> . . . . . 21, 2631	1258, 1259, 1297,	<code>\ifAMC@rbloc</code> . 43, 1008, 1011
<code>\exemplairepair</code> . . . . . 2365	1321, 2331, 2337,	<code>\ifAMC@shuffleG</code> 33, 746, 749
<code>\explain</code> . . . . . 1244, 2520	2374, 2379, 2491, 2605	<code>\ifAMC@survey</code> . . . 2433, 2458
<code>\explaincontext</code> . . . . . 1251	<code>\ifAMC@catalog</code> . . . . . 47, 387	<code>\ifAMC@textPos</code> . 39, 1028,
<code>\ExplSyntaxOff</code> . . . . .	<code>\ifAMC@correc</code> . . . . . 38,	1031, 1041, 1205, 1214
. . . 950, 1402, 1519,	379, 1053, 1245, 1252,	<code>\ifAMC@watermark</code> . . . . .
2036, 2049, 2067, 2557	1620, 2110, 2126, 2493	. . . . . 51, 2267, 2591
<code>\ExplSyntaxOn</code> . . . . .	<code>\ifAMC@correchead</code> . . . . .	<code>\ifAMC@zoneformulaire</code> . .
. . . 864, 1270, 1416,	. . . . . 35, 2267,	. . . . . 57, 527, 564, 868
1524, 2041, 2051, 2538	2310, 2326, 2586, 2592	<code>\ifAMC@complete@multi</code> 44, 841
<b>F</b>	<code>\ifAMC@draw@cross</code> . . . . .	<code>\ifAMC@formulaire@dedans</code>
<code>\fancypagestyle</code> . . . . .	. . . . . 363, 402, 423	. . . . . 56, 526,
. 2269, 2274, 2296, 2303	<code>\ifAMC@ensemble</code> . . . . .	567, 583, 1853, 2020,
<code>\fbox</code> . . . . . 2294	. . . . . 54, 525, 542,	2118, 2175, 2194, 2211
<code>\footrulewidth</code> . . . . .	563, 582, 867, 979,	<code>\ifAMC@ids@side</code> . . . . 75, 2285
. 2272, 2291, 2301, 2306	982, 986, 991, 1207,	<code>\ifAMC@ids@top</code> . . . . 74, 2282
<code>\foreach</code> . . . . . 2484	1409, 1853, 2019,	<code>\ifAMC@needs@storebox</code> . . .
<code>\formulaire</code> . . . . . 21, 2637	2020, 2118, 2123,	. . . . . 2429, 2436, 2447
<code>\fp</code> . . . . . 1430, 1438,	2137, 2143, 2175,	<code>\ifAMC@questionNumber</code> 45, 1172
1440, 1442, 1446,	2194, 2199, 2202,	<code>\ifAMC@sz@logged</code> . . . 580, 591
1451, 1453, 1455,	2211, 2335, 2601, 2611	<code>\ifAMC@type@multi</code> . . . 50, 841
1459, 1466, 1471,	<code>\ifAMC@extractOnly</code> . . . . .	<code>\ifAMC@une@bonne</code> . . . . 49, 843
1478, 1496, 1499,	. . . . . 40, 1987, 2615	<code>\ifcase</code> . . . . . 79
1841, 1842, 1880,	<code>\ifAMC@fullGroups</code> . . 34, 755	<code>\ifcsname</code> . . . 1103, 1108,
1881, 1890, 1918,	<code>\ifAMC@inside@box</code> . . . . .	1113, 1148, 1153, 1158
1919, 1945, 2053,	. . . . . 52, 545, 573,	<code>\ifcsvstrcmp</code> . . . . .
2054, 2056, 2057,	2138, 2200, 2602, 2612	. 2471, 2477, 2478, 2486
2058, 2059, 2061, 2062	<code>\ifAMC@inside@digit</code> . 55, 479	<code>\ifdim</code> . . . . . 1077
<code>francais (option)</code> . . . . . 10	<code>\ifAMC@invisible</code> . . . 61, 293	<code>\ifKV@AMC@BoxOnly@ordered</code>
<code>fullgroups (option)</code> . . . . . 11	<code>\ifAMC@keepmemory</code> 989, 2341	. . . . . 2203
<b>G</b>	<code>\ifAMC@keyline</code> 37, 1206, 1207	<code>\ifKV@AMC@dim@cross</code> . . . . .
<code>\group</code> . . . . . 1422, 1434	<code>\ifAMC@ordre</code> . . . . . 32, 827	. . . . . 383, 398, 415
<b>H</b>	<code>\ifAMC@outside@box</code> . . . . .	<code>\ifKV@AMC@Open@annotate</code> 2153
<code>\he@dbas</code> . . . 2251, 2276, 2277	53, 525, 543, 2603, 2613	<code>\ifKV@AMC@Open@dots</code> . . . 2111
<code>\he@dhaut</code> . . . . . 2252, 2280	<code>\ifAMC@pagelayout</code> . . . . .	<code>\ifKV@AMC@Open@lineup</code> . . .
<code>\he@dtaille</code> . 2248, 2251, 2252	. . . 58, 2264, 2309, 2585	. . . . . 2122, 2165
<code>\hfuzz</code> . . . . . 370	<code>\ifAMC@pdfform</code> . . . . .	<code>\ifKV@AMC@Open@retick</code> . . 2152
<code>\ht</code> . . . . . 368, 1060	62, 351, 2329, 2346, 2439	<code>\ifKV@AMC@Open@scan</code> . . . 2151
<b>I</b>	<code>\ifAMC@plain</code> . . . . . 48, 2522	<code>\ifSurveySingleAnswer</code> . .
<code>\ifAMC@affichekeys</code> . . . . .	<code>\ifAMC@postcorrect</code> . . . . .	. . . . . 1220, 1223, 1229
. . . . . 36, 1206, 1207	. . . . . 59, 844, 2604, 2614	<code>indivanswers (option)</code> . 6, 10
<code>\ifAMC@asqbloc</code> . 42, 847, 853	<code>\ifAMC@printformoutside</code>	<code>init (option)</code> . . . . . 10
<code>\ifAMC@automarks</code> . . . . .	. . . . . 523, 532	<code>\InputIfFileExists</code> . . . . 15
60, 979, 2335, 2354, 2361	<code>\ifAMC@printformoutside@</code>	<code>\insertgroup</code> . . 14, 745, 2635
<code>\ifAMC@calibration</code> . . . . .	. . . . . 522, 529	<code>\insertgroupfrom</code> . . . 14, 745
. . . . . 46, 295, 304,	<code>\ifAMC@printkeyoutside</code> .	<code>insidebox (option)</code> . . . . . 10
313, 326, 331, 332,	. . . . . 540, 550	<code>\int</code> . . . . . 879, 881, 885,
	<code>\ifAMC@printkeyoutside@</code>	891, 898, 937, 1274,
	. . . . . 539, 547	1287, 1326, 1328,
		1337, 1343, 1344,



<code>\seq</code> . . . . .	1273, 1347, 1355, 1421, 1429, 1430, 1431	<code>\textsc</code> . . . . .	2270	<code>\usetikzlibrary</code> . . . . .	2496
<code>\setdefaultgroupmode</code> . .	<u>687</u>	<code>\textsf</code> . . . . .	505		
<code>\setgroupmode</code> . . . . .	673, <u>687</u>	<code>\thecsvrow</code> . . . . .		<b>V</b>	
<code>\shuffle@it</code> . . . . .	<u>819</u> , 833	. . . . .	1033, 1034, 2499, 2508	<code>values (env.)</code> . . . . .	<u>2458</u>
<code>\shufflegroup</code> . . . . .		<code>tikz-multi (env.)</code> . . . . .	<u>1005</u>	<code>values-auto (env.)</code> . . . . .	<u>2458</u>
. . . . .	14, 711, 723, <u>745</u> , 2634	<code>tikz-single (env.)</code> . . . . .	<u>1005</u>	<code>variable-auto (env.)</code> . . .	<u>2458</u>
<code>\shufflegroupslice</code> . . . . .		<code>\tikzset</code> . . . . .	2497	<code>variable-multi (env.)</code> . .	<u>1179</u>
. . . . .	728, 731, 748	<code>\times</code> . . . . .	1532	<code>variable-single (env.)</code> .	<u>1179</u>
<code>\smash</code> . . . . .	2110, 2126	<code>\tl</code> . . . . .	899, 900, 905, 910, 939, 941, 1418, 1420, 1427, 1428, 1629, 1632, 1638, 1641, 1792, 1806, 1808, 1811, 1812, 1816, 1821, 1822, 1824, 1827, 1828, 1836, 1848, 1849	<code>\version</code> . . . . .	2608
<code>\space</code> . . . . .	2520			<code>\vfuzz</code> . . . . .	370
<code>\storebox</code> . . . . .	2449			<code>\vrule</code> . . . . .	1778
<code>storebox (option)</code> . . . . .	11			<code>\vtop</code> . . . . .	1350
<code>\str</code> . . . . .	934				
<code>\strut</code> . . . . .	1078, 1080			<b>W</b>	
<code>\SujetExterne</code> . . . . .	<u>2558</u>			<code>\wd</code> . . . . .	370, 371, 1060
<code>\SurveySingleAnswerfalse</code>		<code>\tracepos</code> . . . . .	297, 306, 315	<code>\with</code> . . . . .	1322, 2380, 2381, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2618, 2620
. . . . .	1220			<code>\wrongchoice</code> . . . . .	13, 843, 844, <u>1050</u> , 2060, 2627
		<b>U</b>			
<b>T</b>		<code>\une@rep</code> . . . . .	1007, 1008, 1013, 1017, 1021, 1024, 1053, 1057, 2493	<b>X</b>	
<code>\tex</code> . . . . .	1424			<code>\x</code> . . . . .	2484, 2486
<code>\textasciicircum</code> . . . . .	1532	<code>\useasboundingbox</code> . . . . .	418	<code>\xyclose</code> . . . . .	345
<code>\TextField</code> . . . . .	353	<code>\usebox</code> . . . . .	362	<code>\xyopen</code> . . . . .	343
<code>\textit</code> . . . . .	100, 113, 140, 154, 193, 219	<code>\usestorebox</code> . . . . .	2450		