

# The automultiplechoice package\*

Alexis Bienvenue  
paamc@passoire.fr

March 7, 2018

## 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}
```

---

\*This document corresponds to version revision: r:bf07b77e8a6d from AMC 1.3.0+hg2018-03-07

```

        \wrongchoice{Addis Abeba}
        \wrongchoice{Ankara}
        \wrongchoice{Apia}
    \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}
}

```

```

\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 8.

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:



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 9. 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 10.

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


\*1/1/50\*

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 `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?

☐ 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  
☐ Cairo  
☐ 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  
☐ Kwing-en 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`.


\*2/1/50\*

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 `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?

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

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

☐ Djibouti ☐ Dhaka ☐ Doha  
☐ Dublin ☐ 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  
☐ Kwing-en becomes emperor of China

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


\*3/1/50\*

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 `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?

☐ 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  
☐ Kwing-en 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 ☐ Doha ☐ Dakar  
☐ Dili ☐ Dublin ☐ Djibouti

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

☐ Beijing ☐ Banja ☐ Bangkok ☐ Beirut ☐ Berlin

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


\*4/1/50\*

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 `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?

☐ 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?

☐ Cayenne  
☐ Caracas  
☐ Cairo  
☐ 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  
☐ Kwing-en becomes emperor of China

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


☐ Djibouti ☐ Dhaka ☐ Dakar  
☐ Dili ☐ Doha ☐ 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

[illegible]



+1/2/5/+

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

DRAFT



For your examination, preferably print documents compiled from auto-multiple-choice.



-2/1/80-

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. DIAGPT indications can be cancelled using *seventrans* 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 1801?

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

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

- ☐ Dýghouði
- ☐ Dýshla
- ☐ Dýblu
- ☐ Dýla


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

- ☐ Apia
- ☐ Accra
- ☐ Abba Alaba
- ☐ Ankara

**Question 4 ▲** Which of the following events are taking place during the year 1807?

- ☐ Napoleon Bonaparte crosses the Alps with Grande Armée
- ☐ First horse-drawn omnibuses established in London
- ☐ American Express is founded by Henry Wells & William Fargo
- ☐ Average-on becomes emperor of China

For your examination, preferably print documents compiled from *automultiplechoice*.



•2/2/37•

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	E	F
---	---	---	---	---	---

Question 3: 

A	B	C	D
---	---	---	---

Question 4: 

A	B	C	D
---	---	---	---

DRAFT

For your examination, preferably print documents compiled from auto-multiple-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:

- `nosage`, 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 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

☒ Cairo

☐ 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-su 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:

- `nosage`, 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 Métro is opened

☒ Official end of the Caste War of Yucatán

☐ King George of Greece becomes absolute monarch of Crete

☒ Funeral of Queen Victoria in London

Question 2 What is the capital of Ireland?

☐ Džibouti

☐ Dhaka

☐ Doha

☒ Dublin

☐ 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-su becomes emperor of China

1


AMC

SAMPLE TEST

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

- `nosage`, 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 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-su 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?

☐ Doha

☐ Doha

☒ Dublin

☐ Dili

☐ Džibouti

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:

- `nosage`, 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 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

☒ Cairo

☐ 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-su becomes emperor of China

Question 4 What is the capital of Ireland?

☐ Džibouti

☐ Dhaka

☐ Doha

☒ Dublin

☐ Dili

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 questions identifiers will be printed.

`français` asks for french localisation.

`lang=XX` asks for localisation in `XX` language. At present, only `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.

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

```

\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}

```

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

- ☐ 8,253 m
- ☐ 8,810 m
- ☐ 8,848 m

```

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

```

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

- ☐ Cambodia
- ☐ Guatemala
- ☐ Canada
- ☐ Switzerland

\AMCcompleteMulti  
MCnoCompleteMulti

For multiple questions, it is sometimes useful to make the difference between a student who thinks that none of the choices are correct, and a student who did not answer the question. 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.

```

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

```

**Question 1 ♣** Which contries are in the Americas?

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

choices  
choiceshoriz  
choicescustom

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

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

```

\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}

```

**Question 1 ♣** 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.

- environment `choiceshoriz` is chosen for short answers:

```

\begin{question}{insect}
  From those animals, which
  is an insect?
  \begin{choiceshoriz}
    \correctchoice{Ant}
    \wrongchoice{Horse}
    \wrongchoice{Turtle}
  \end{choiceshoriz}
\end{question}

```

**Question 1** From those animals, which is an insect?

- ☐ Horse      ☐ Ant      ☐ Turtle

- 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 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. 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, consider the following code:

```
\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:\insertgroup[2]{serie}.
```

which produces:

Numbers: one two three four five. Three numbers from the second (index=1) one: two three four. Two of them: two four.
---

`\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. 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, consider the following code:

```
\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}
```





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

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

0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

### 3.6 Separate answer sheet

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

`\AMCform`  
`\MCcleardoublepage`

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`):

```
\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}
```

<b>Question 1</b>	How much are 2 plus 8?
<input type="checkbox"/> 9	<input checked="" type="checkbox"/> 10
<input type="checkbox"/> 16	<input type="checkbox"/> -6

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{⟨x⟩}{⟨x0⟩}{⟨x1⟩}{⟨delta⟩}` 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.

```

\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{reponses}[o]
      \AMCIntervals{\VQr}{0}{1}{0.1}
    \end{reponses}
  \end{multicols}
\end{question}

```

### Question 1

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[   |

AMCnumericChoices

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

```

\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}

```

### Question 2

Compute  $\sqrt{11}$  and round it up to 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

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 (*bool* can be `true` or `false`, and *color* must be a color known by the `xcolor` package):

`digits=num` gives the number of digits to request (defaults to 3).

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

`significant=bool` 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=num` gives the number of digits for the exponent, when requesting to enter the result in scientific notation.

`nozero=bool` 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=bool` requests (or not) a signed value (default to `true`).

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

`strict=bool` 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=bool` if `true`, each digit is represented on one raw. If `false` (default), each digit is represented on one line.

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

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

`vhead=bool` 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`.

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

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

`borderwidth=space` 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\AMCncontextSign{text}`, and defaults to be empty).

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

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

**scoring**= $\langle bool \rangle$  if **true**, a scoring strategy is given to AMC for this question. Defaults to **true**.

**scoreexact**= $\langle num \rangle$  gives the score for an exact answer (defaults to 2).

**exact**= $\langle num \rangle$  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**= $\langle num \rangle$  gives the score for an approximative answer (defaults to 1).

**approx**= $\langle num \rangle$  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).

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 `\AMCncontextGoto` command, as:

```
\def\AMCncontextGoto{\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{\langle style \rangle}` can be used to specify the shape, color and dimensions of the boxes to be ticked. The argument  $\langle style \rangle$  is a coma-separated list of  $\langle key \rangle = \langle value \rangle$  pairs, with the following possible  $\langle key \rangle$ 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.

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

`outsidesep` 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,outsidesep=.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).

```

\AMCboxStyle{shape=oval,color=red}
\begin{question}{sum}$2+2={}$
\begin{choiceshoriz}[o]
  \wrongchoice{1}\correctchoice{4}\wrongchoice{10}
\end{choiceshoriz}
\end{question}

```

**Question 1**     $2 + 2 =$

☐ 1      ☐ 4      ☐ 10

### 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{<box>}{<text>}` (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...

<pre> \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} </pre>	<div style="border-left: 1px solid black; padding-left: 10px;"> <p><b>Question 1</b>      <math>2+2= \left( \boxed{4} \quad \boxed{3} \quad \boxed{2} \right)</math></p> </div>
--	---

## 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}

```

`\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.

```

8 \newwrite\AMC@logfile
9 \immediate\openout\AMC@logfile=\jobname.amc
10 \def\AMC@amclog#1{\immediate\write\AMC@logfile{#1}}
11 \def\AMCmessage#1{\AMC@amclog{AUTOQCM[#1]^^J}}

```

`\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.

```

12 \AtBeginDocument{\@ifpackageloaded{bidi}{%
13   \PackageInfo{automultiplechoice}{Package bidi loaded: using LR for boxes.}%
14   \let\AMC@LR=\LR}%
15 {\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@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).

`\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  $\text{\LaTeX}$  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.

```

16 \newcount\AMCload@counter
17 \newcount\AMCid@quest\AMCid@quest=-1
18 \newcount\AMCid@check
19 \newcount\AMCid@etud\AMCid@etud=0
20 \newcount\AMCid@etudstart\AMCid@etudstart=0
21 \newcount\AMCid@etudfin
22 \newcount\AMCnum@copies

23 \newif\ifAMC@ordre\AMC@ordrefalse
24 \newif\ifAMC@shuffle\AMC@shuffleGtrue
25 \newif\ifAMC@fullGroups\AMC@fullGroupsfalse
26 \newif\ifAMC@correthead\AMC@corretheadfalse
27 \newif\ifAMC@affichekeys\AMC@affichekeysfalse
28 \newif\ifAMC@correc\AMC@correcfalse
29 \newif\ifAMC@qbloc\AMC@qblocfalse
30 \newif\ifAMC@asqbloc\AMC@asqblocfalse
31 \newif\ifAMC@rbloc\AMC@rblocfalse
32 \newif\ifAMC@complete@multi\AMC@complete@multifalse
33 \newif\ifAMC@questionNumber\AMC@questionNumbertrue
34 \newif\ifAMC@calibration\AMC@calibrationfalse
35 \newif\ifAMC@plain\AMC@plainfalse
36 \newif\ifAMC@bonne
37 \newif\ifAMCtype@multi
38 \newif\ifAMC@watermark\AMC@watermarktrue

```



```

39 \newif\ifAMC@inside@box\AMC@inside@boxfalse
40 \newif\ifAMC@outside@box\AMC@outside@boxfalse
41 \newif\ifAMC@ensemble\AMC@ensemblefalse
42 \newif\ifAMC@inside@digit\AMC@inside@digitfalse
43 \newif\ifAMC@formulaire@dedans\AMC@formulaire@dedansfalse
44 \newif\ifAMC@zoneformulaire
45 \newif\ifAMC@pagelayout\AMC@pagelayouttrue
46 \newif\ifAMC@postcorrect\AMC@postcorrectfalse
47 \newif\ifAMC@automarks\AMC@automarksfalse
48 \newif\ifAMC@invisible\AMC@invisiblefalse
49 \newif\ifAMC@pdfform\AMC@pdfformfalse
50 \let\AMCcompleteMulti=\AMCcomplete@multitrue
51 \let\AMCnoCompleteMulti=\AMCcomplete@multifalse

```

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

```
52 \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.

`\AMCinterIrep` `\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`).

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

```

53 \newdimen\AMCformVSpace\AMCformVSpace=1.2ex
54 \newdimen\AMCformHSpace\AMCformHSpace=.3em
55 \newdimen\AMCinterIrep\AMCinterIrep=\z@
56 \newdimen\AMCinterBrep\AMCinterBrep=.5ex
57 \newdimen\AMCinterIquest\AMCinterIquest=\z@
58 \newdimen\AMCinterBquest\AMCinterBquest=3ex
59 \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).

```

60 \newif\ifAMCids@top
61 \newif\ifAMCids@side
62 \newdimen\AMCids@width
63 \newdimen\AMCids@height
64 \define@choicekey*{AMCids}{pos}[\AMCidsVar\AMCidsVarN]{none,top,side}{%
65   \ifcase\AMCidsVarN\relax
66     \AMCids@topfalse\AMCids@sidefalse
67   \or
68     \AMCids@toptrue\AMCids@sidefalse
69   \or
70     \AMCids@topfalse\AMCids@sidetrue
71   \fi
72 }
73 \define@key{AMCids}{width}{\AMCids@width=#1}
74 \define@key{AMCids}{height}{\AMCids@height=#1}
75 \def\AMCidsPosition#1{\setkeys{AMCids}{#1}}
76 \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.

```

77 \def\AMCtext#1#2{\expandafter\def\csname AMC@loc@#1\endcsname{#2}}
78 \def\AMClocalized#1{\csname AMC@loc@#1\endcsname}

```

#### 4.4.1 English

Text indicating draft exams:

```
79 \def\AMC@loc@draft{DRAFT}
```

Message at page bottom when compiled out of AMC gui:

```

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

```

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

```
82 \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):

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

Headers for corrected version and catalog:

```
84 \def\AMC@loc@corrected{Corrected}
85 \def\AMC@loc@catalog{Catalog}
```

Localization text for Explanation

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

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

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

Word for 'question', singular and plural forms:

```
88 \def\AMC@loc@question{question}
89 \def\AMC@loc@questions{questions}
```

Default text to write in the students' name box:

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

#### 4.4.2 Dutch

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

```
91 \def\AMC@loc@NL{
92   \def\AMC@loc@draft{Ontwerp}
93   \def\AMC@loc@message{Gebruik bij uw proefwerk bij voorkeur die
94     documenten welke door auto-multiple-choice zijn aangemaakt.}
95   \def\AMC@loc@qf##1{\textbf{Vraag ##1 :}}
96   \def\AMC@loc@q##1##2{\textbf{Vraag ##1} ##2}
97   \def\AMC@loc@corrected{Correctie}
98   \def\AMC@loc@catalog{Catalogus}
99   \def\AMC@loc@none{Geen van de antwoorden is juist.}
100  \def\AMC@loc@question{vraag}
101  \def\AMC@loc@questions{vragen}
102 }
```

#### 4.4.3 French

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

```
103 \def\AMC@loc@FR{
104   \def\AMC@loc@draft{PROJET}
105   \def\AMC@loc@message{Pour votre examen, imprimez de pr\'ef\'erence
106     les documents compil\'es \'a l'aide de auto-multiple-choice.}
107   \def\AMC@loc@qf##1{\textbf{Question ##1 :}}
108   \def\AMC@loc@q##1##2{\textbf{Question ##1} ##2}
109   \def\AMC@loc@corrected{Correction}
110   \def\AMC@loc@catalog{Catalogue}
111   \def\AMC@loc@explain{\textit{\textbf{Explication : }}}}
112   \def\AMC@loc@none{Aucune de ces r\'eponses n'est correcte.}
113   \def\AMC@loc@question{question}
114   \def\AMC@loc@questions{questions}
115   \def\AMC@loc@namesurname{Nom et pr\'enom :}
116 }
```

#### 4.4.4 German

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

```
117 \def\AMC@loc@DE{
118   \def\AMC@loc@draft{ENTWURF}
119   \def\AMC@loc@message{Benutzen Sie f\"ur Ihre Pr\"ufung bevorzugt Dokumente die mit
120     auto-multiple-choice erstellt wurden.}
121   \def\AMC@loc@qf##1{\textbf{Frage ##1 :}}
122   \def\AMC@loc@q##1##2{\textbf{Frage ##1} ##2}
123   \def\AMC@loc@corrected{Korrektur}
124   \def\AMC@loc@catalog{Katalog}
125   \def\AMC@loc@explain{\textit{\textbf{Erkl\"arung : }}}
126   \def\AMC@loc@none{Keine dieser Antworten ist korrekt.}
127   \def\AMC@loc@question{Frage}
128   \def\AMC@loc@questions{Fragen}
129 }
```

#### 4.4.5 Italian

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

```
130 \def\AMC@loc@IT{
131   \def\AMC@loc@draft{BOZZA}
132   \def\AMC@loc@message{Per l'esame, \e preferibile stampare i documenti
133     a partire da auto-multiple-choice.}
134   \def\AMC@loc@qf##1{\textbf{Domanda ##1:}}
135   \def\AMC@loc@q##1##2{\textbf{Domanda ##1} ##2}
136   \def\AMC@loc@corrected{Correzione}
137   \def\AMC@loc@catalog{Catalogo}
138   \def\AMC@loc@none{Nessuna risposta \e giusta.}
139   \def\AMC@loc@question{domanda}
140   \def\AMC@loc@questions{domande}
141 }
```

#### 4.4.6 Norwegian

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

```
142 \def\AMC@loc@NO{
143   \def\AMC@loc@draft{UTKAST}
144   \def\AMC@loc@message{Det anbefales {\aa} skrive ut dokumentet
145     for gjennomgang \direkte fra auto-multiple-choice.}
146   \def\AMC@loc@qf##1{\textbf{Oppgave ##1 :}}
147   \def\AMC@loc@q##1##2{\textbf{Oppgave ##1} ##2}
148   \def\AMC@loc@corrected{Rettet}
149   \def\AMC@loc@catalog{Katalog}
150   \def\AMC@loc@none{Ingen svar er riktige.}
151   \def\AMC@loc@question{oppgave}
152   \def\AMC@loc@questions{oppgave}
153 }
```

#### 4.4.7 Portuguese

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

```
154 \def\AMC@loc@PT{
155   \def\AMC@loc@draft{RASCUNHO}
156   \def\AMC@loc@message{Para o seu exame, use preferencialmente documentos compilados do auto-multiple-choice}
157   \def\AMC@loc@qf##1{\textbf{Quest\~ao ##1:}}
158   \def\AMC@loc@q##1##2{\textbf{Quest\~ao ##1} ##2}
159   \def\AMC@loc@corrected{Corrigido}
160   \def\AMC@loc@catalog{Cat\'alogo}
161   \def\AMC@loc@explain{\textit{\textbf{Justifique: }}}
162   \def\AMC@loc@none{Nenhuma das respostas apresentadas est\'a correta.}
163   \def\AMC@loc@question{Quest\~ao}
164   \def\AMC@loc@questions{Quest\~oes}
165 }
```

#### 4.4.8 Spanish

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

```
166 \def\AMC@loc@ES{
167   \def\AMC@loc@draft{BORRADOR}
168   \def\AMC@loc@message{Para revisi\'on, preferentemente imprimir documentos compilados
169     desde auto-multiple-choice.}
170   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
171   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
172   \def\AMC@loc@corrected{Correcci\'on}
173   \def\AMC@loc@catalog{Cat\'alogo}
174   \def\AMC@loc@none{Ninguna de estas preguntas son correctas.}
175   \def\AMC@loc@question{pregunta}
176   \def\AMC@loc@questions{preguntas}
177 }
```

#### 4.4.9 Japanese

Japanese localisation is called with option `lang=JA`. It includes UTF8 encoded Japanese characters which are shown as  $\diamond$  here (look at the `.sty` file to see them).

```
178 \def\AMC@loc@JA{
179   \def\AMC@loc@draft{\diamond\diamond\diamond}
180   \def\AMC@loc@message{\diamond\diamond\diamondauto-multiple-choice\diamond\diamond\diamond\diamond\diamond\diamond}
181   \def\AMC@loc@qf##1{\textbf{\diamond##1:}}
182   \def\AMC@loc@q##1##2{\textbf{\diamond##1} ##2}
183   \def\AMC@loc@corrected{\diamond\diamond\diamond}
184   \def\AMC@loc@catalog{\diamond\diamond\diamond}
185   \def\AMC@loc@explain{\textit{\textbf{\diamond: }}}
186   \def\AMC@loc@none{\diamond\diamond\diamond}
187   \def\AMC@loc@question{\diamond}
188   \def\AMC@loc@questions{\diamond}
189 }
```

#### 4.4.10 Other languages

Other languages can be integrated to automultiplechoice package upon request to the author.

### 4.5 Interaction with other packages

#### 4.5.1 cleveref

For references to questions:

```
190 \AtBeginDocument{\@ifpackageloaded{cleveref}{%
191   \message{AMC/cleveref integration loaded^^J}%
192   \creffalias{AMCquestionaff}{question}%
193   \crefname{question}{\AMC@loc@question}{\AMC@loc@questions}%
194 }{}}%
```

### 4.6 Random

#### 4.6.1 Random pseudo-generator

The package uses the pseudo-random bit generator from *TuGBoat* 1994, vol 15:1:

```
195 \ifx\AMC@SR\undefined\newcount\AMC@SR\fi
196 \providecommand\AMC@SRconst{2097152}
197 \providecommand\AMC@SRset[1]{\global\AMC@SR#1 \ignorespaces}
198 \providecommand\AMC@SRadvance{%
199   \begingroup%
200     \ifnum\AMC@SR<\AMC@SRconst\relax\AMC@SR@count\z@ \else\AMC@SR@count\@ne\fi%
201     \ifodd\AMC@SR\advance\AMC@SR@count\@ne\fi%
202     \global\divide\AMC@SR\tw@%
203     \ifodd\AMC@SR@count\global\advance\AMC@SR\AMC@SRconst\relax\fi%
204   \endgroup}
205 \providecommand\AMC@SRbit{\AMC@SRadvance\ifodd\AMC@SR1\else0\fi}
206 \providecommand\AMC@SRtest[2]{\AMC@SRadvance%
207   \ifodd\AMC@SR#2\else#1\fi\ignorespaces}
208 \providecommand\AMC@SRvalue{\number\AMC@SR}
```

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

```
209 \AMC@SRset{1515}
210 \def\AMCrandomseed#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.

```
211 \newcount\AMC@SR@count
212 \def\AMC@SR@time{\AMC@SRset{\time}}
213 \newcount\AMC@SRnum
214 \def\AMC@SRnextByte{\AMC@SRnum=\z@%
215   \AMC@SR@count=20%
```

```

216 \loop\multiply\AMC@SRnum\tw%
217 \AMC@SRtest{\advance\AMC@SRnum\@ne}{}%
218 \ifnum\AMC@SR@count>\@ne\advance\AMC@SR@count\m@ne\repeat%
219 }
220 \newcommand\AMC@SRmax[1]{\AMC@SRnextByte%
221 \AMC@SR@count=\AMC@SRnum%
222 \divide\AMC@SR@count by #1\relax%
223 \multiply\AMC@SR@count by #1\relax%
224 \advance\AMC@SRnum by -\AMC@SR@count%
225 }

```

### 4.6.3 Tokens shuffling

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

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  $\langle a \rangle$ , registers are shuffled from number  $\langle a \rangle$  to  $\langle n \rangle$  (default value for  $\langle a \rangle$  is 1).

```

226 \newcount\AMC@sti
227 \newcount\AMC@stil
228 \newtoks\AMCsw@p@
229 \newcommand\AMCsw@p[2]{%
230 \global\AMCsw@p@=#1%
231 \global#1=#2%
232 \global#2=\AMCsw@p@}
233 \newcommand{\AMC@shuffletoks}[3][\@ne]{%
234 \AMC@sti=#2\relax%
235 \AMC@stil=#2\relax%
236 \advance\AMC@stil\@ne%
237 \advance\AMC@stil -#1\relax%
238 \@whilenum\AMC@sti>#1\do{%
239 \AMC@SRmax{\AMC@stil}\advance\AMC@SRnum #1\relax%
240 \AMCsw@p{\csname #3\romannumeral\AMC@SRnum\endcsname}%
241 {\csname #3\romannumeral\AMC@sti\endcsname}%
242 \advance\AMC@sti\m@ne\relax%
243 \advance\AMC@stil\m@ne\relax%
244 }}

```

## 4.7 Keys numbering

`\AMC@unnumero` This package allocates a unique integer ID to each question key from the questionnaire. The  
`\AMC@affecte` 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).

```

245 \newcount\AMC@numerotation\AMC@numerotation=\z@%
246 \def\AMC@definitnumero#1#2{\AMC@amclog{AUTOQCM[ $\text{NUM}=\text{#1}=\text{#2}$ ]^^J}}%

```

```

247 \expandafter\global\expandafter\def\csname AMC@numtab@#2\endcsname{#1}}
248 \def\AMC@prepare#1{\expandafter\ifx\csname AMC@numtab@#1\endcsname\relax%
249 \global\advance\AMC@numerotation\@ne%
250 \expandafter\AMC@definitnumero\expandafter{\the\AMC@numerotation}{#1}\fi}
251 \def\AMC@unnumero#1{\AMC@prepare{#1}\csname AMC@numtab@#1\endcsname}
252 \def\AMC@affecte#1#2{\AMC@prepare{#1}\global#2=\csname AMC@numtab@#1\endcsname}

```

## 4.8 Boxes

### 4.8.1 Position logging

`\AMC@tracebox` Command `\AMC@tracebox{<trace>}{<key>}{<content>}` makes a L<sup>A</sup>T<sub>E</sub>X box around `<content>`, 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.

```

253 \def\AMC@shapename@{\ifAMC@invisible none\else\AMC@shapename\fi}
254 \def\AMC@tracepos#1#2{%
255 \ifAMC@calibration\ifx\@empty#1\@empty\else%
256 \pdfsavepos\protected@write\AMC@XYFILE{ }{%
257 \string\tracepos%
258 {\the\AMCid@etud/\thepage:#2}%
259 {\noexpand\number\pdflastxpos sp}%
260 {\noexpand\number\pdflastypos sp}%
261 {\AMC@shapename}}%
262 \fi\fi}
263 \def\AMC@traceposx#1#2{%
264 \ifAMC@calibration\ifx\@empty#1\@empty\else%
265 \pdfsavepos\protected@write\AMC@XYFILE{ }{%
266 \string\tracepos%
267 {\the\AMCid@etud/\thepage:#2}%
268 {\noexpand\number\pdflastxpos sp}%
269 {0sp}%
270 {\AMC@shapename}}%
271 \fi\fi}
272 \def\AMC@traceposy#1#2{%
273 \ifAMC@calibration\ifx\@empty#1\@empty\else%
274 \pdfsavepos\protected@write\AMC@XYFILE{ }{%
275 \string\tracepos%
276 {\the\AMCid@etud/\thepage:#2}%
277 {0sp}%
278 {\noexpand\number\pdflastypos sp}%
279 {\AMC@shapename}}%
280 \fi\fi}
281 \newcommand\AMC@tracebox[3]{%
282 \vbox{\AMC@traceposy{#1}{#2}%
283 \hbox{\AMC@traceposx{#1}{#2}#3\AMC@traceposx{#1}{#2}}%
284 \AMC@traceposy{#1}{#2}}
285 \def\AMC@pagepos{%
286 \ifAMC@calibration\protected@write\AMC@XYFILE{ }{%
287 \string\page%

```



```

288   {\the\AMCid@etud/\thepage/\the\AMCid@check}%
289   {\the\paperwidth}{\the\paperheight}}\fi}

```

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

```

290 \newcommand{\AMCdontScan}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontscan{\the\AMCid@etud,\th
291 \newcommand{\AMCdontAnnotate}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontannotate{\the\AMCid@
292 %

```

`\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.

```

293 \newcommand\AMC@tracechar[4]{%
294   \ifAMC@calibration\ifx\@empty#3\@empty\else%
295     \protected@write\AMC@XYFILE{}{%
296       \string\boxchar{\the\AMCid@etud/\thepage:#4}{#1}%
297     }%
298   \fi\fi%
299 }

```

`amcxyfile`    The following lines defines an environment to use a particular file for positions outputs. This is used mainly for documentation or testing.

```

300 \newwrite\AMC@XYspecial
301 \newwrite\AMC@tmpXY
302 \newenvironment{amcxyfile}[1]{%
303   \openout\AMC@XYspecial#1%
304   \let\AMC@tmpXY=\AMC@XYFILE%
305   \let\AMC@XYFILE=\AMC@XYspecial%
306 }{\let\AMC@XYFILE=\AMC@tmpXY\closeout\AMC@XYspecial}

```

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

```

307 \newcommand{\namefield}[1]{\AMC@tracebox{1}{nom}{#1}}

```

It is used to enclose the page region where students are to write their names, so as te 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:

```

308 \newcommand{\namefielddots}{%
309   \noindent%
310   \ifAMC@pdfform%
311     \hspace*{\fill}%
312     \TextField[name={\the\AMCid@etud:namefield},width=.95\linewidth,bordercolor=0 0 0]{}%
313     \hspace*{\fill}
314   \else%
315     \dotfill
316   \fi%
317 }

```

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.

#### 4.8.2 Boxes to be checked by students

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@checkedbox` 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 

K
---

, writing the lines in the .xy file shown in section 5.2.
- `\AMC@answerBox@{K}{\AMC@checkedbox}{}` produces 

--
- `\AMC@answerBox@{}{8}{}` produces 

8
---
- `\AMC@answerBox@{K}{8}{1}{testb}` produces 

8
---

 with `\AMCboxStyle{shape=oval,color=red}`

```

318 \def\AMC@checkedbox{}
319 \let\AMC@new@savebox=\newsavebox
320 \let\AMC@save@box=\savebox
321 \let\AMC@use@box=\usebox
322 \newif\ifAMC@draw@cross

```

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

```

323 \newbox\AMC@smashbox
324 \newdimen\AMC@smashboxheight

```

```

325 \newcommand{\AMC@smashcentered}[1]{%
326   \setbox\AMC@smashbox\hbox{#1}%
327   \AMC@smashboxheight=\ht\AMC@smashbox%
328   \advance\AMC@smashboxheight by \dp\AMC@smashbox%
329   \vfuzz=\AMC@smashboxheight\hfuzz=\wd\AMC@smashbox%
330   \hspace*{-.5\wd\AMC@smashbox}\hbox to .5\wd\AMC@smashbox{%
331     \vbox to 0pt{%
332       \vspace*{-.5\AMC@smashboxheight}\vbox to .5\AMC@smashboxheight{%
333         \box\AMC@smashbox}}}%
334   }}

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

335 \newcommand\AMC@setcolors@[2]{%
336   \def\AMC@boxcolor@{\AMC@boxcolor}%
337   \ifx\@empty#1\@empty \def\AMC@boxcolor@{black}\fi%
338   \ifAMC@correc\def\AMC@boxcolor@{black}\fi%
339   \def\AMC@fillcolor@{\ifx #2\AMC@checkedbox%
340     \AMC@boxcolor@\else white\fi}%
341   \AMC@draw@crossfalse%
342   \ifKV@AMCdim@cross\ifx #2\AMC@checkedbox%
343     \AMC@draw@crosstrue\fi\fi%
344 }

345 \newcommand\AMC@answerBox@[4]{%
346   \AMC@LR{\hspace{0pt}%
347     \lower\AMC@boxeddown\hbox{\csname AMC@shape@\AMC@shapename@\endcsname%
348       {\AMCchoiceLabelFormat{#1}}{#2}{#3}{#4}}}%
349 }

350 \newcommand\AMC@shapeprepare@square{}
351 \newcommand\AMC@shape@square[4]{%
352   \fboxsep=\z@\fboxrule=\AMC@boxedrule%
353   \AMC@setcolors@{#3}{#2}%
354   \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
355   \fcolorbox{\AMC@boxcolor@}{\AMC@fillcolor@}%
356   {%
357     \boxput*(0,0){%
358       \ifAMC@draw@cross\AMC@crosschar\fi%
359     }{%
360       \vbox to \AMC@boxedheight{%
361         \AMC@tracepos{#3}{#4}%
362         \vfill%
363         \hbox to \AMC@boxedwidth{\hfill%
364           \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
365           \AMC@smashcentered{#2}%
366           \hfill}\vfill}}%
367       \AMC@tracepos{#3}{#4}}%
368 }

\AMC@makeovalbox{\trace}{\answer}{\box} prepares an oval frame in the LATEX box \box.

369 \newcommand\AMC@makeovalbox[3]{%

```

```

370 \AMC@setcolors@{#1}{#2}%
371 \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
372 \AMC@save@box{#3}{%
373 \begin{tikzpicture}%
374 \useasboundingbox (-0.5\AMC@boxedwidth-0.5\AMC@boxedrule,0.5\AMC@boxedheight+0.5\AMC@boxedrule)
375 rectangle (0.5\AMC@boxedwidth+0.5\AMC@boxedrule,-0.5\AMC@boxedheight-0.5\AMC@boxedrule);
376 \draw[\AMC@boxcolor@,fill=\AMC@fillcolor@,line width=\AMC@boxedrule,rounded corners=\AMC@oval@radius]
377 (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight)
378 rectangle (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
379 \ifAMC@draw@cross
380 \draw[\AMC@boxcolor@,line width=\AMC@crossrule]
381 (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight)
382 (0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (-0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
383 \fi
384 \end{tikzpicture}}%
385 }
386 \newcommand\AMC@shapeprepare@oval{%
387 \AMC@makeovalbox{1}{\AMC@ovalbox@R}%
388 \AMC@makeovalbox{1}{\AMC@checkedbox}{\AMC@ovalbox@RF}%
389 \AMC@makeovalbox{}{}{\AMC@ovalbox@}%
390 \AMC@makeovalbox{}{}{\AMC@checkedbox}{\AMC@ovalbox@F}%
391 }
392 \newcommand\AMC@shape@oval[4]{%
393 \AMC@setcolors@{#3}{#2}%
394 \AMC@tracebox{#3}{#4}{\boxput*(0,0){%
395 \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
396 \AMC@smashcentered{#2}%
397 }{%
398 \ifx\@empty#3\@empty%
399 \ifx #2\AMC@checkedbox%
400 \AMC@use@box{\AMC@ovalbox@F}%
401 \else%
402 \AMC@use@box{\AMC@ovalbox@}%
403 \fi%
404 \else%
405 \ifx #2\AMC@checkedbox%
406 \AMC@use@box{\AMC@ovalbox@RF}%
407 \else%
408 \AMC@use@box{\AMC@ovalbox@R}%
409 \fi%
410 \fi%
411 }}%
412 }
413 \newcommand\AMC@shapeprepare@form{}
414 \newcommand\AMC@shape@form@base[5]{%
415 \ifx #2\AMC@checkedbox%
416 \def\AMC@shape@form@ticked{true}%
417 \else%
418 \def\AMC@shape@form@ticked{false}%
419 \fi%

```

```

420 \AMC@tracebox{#3}{#4}{%
421   \CheckBox[checked=\AMC@shape@form@ticked,%
422             checkboxsymbol=\ding{110},name={#5},%
423             bordercolor=0 0 0,
424             width=\AMC@boxedwidth,height=\AMC@boxedheight]{}{}%
425 }%
426 }
427 \newcommand\AMC@shape@form[4]{%
428   \AMC@shape@form@base{#1}{#2}{#3}{#4}{\the\AMCid@etud:#4}%
429 }
430 \newcommand\AMC@shapeprepare@none{}
431 \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.

`\AMCchoiceLabelFormat` 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{\textsf{#1}}}`.

```

432 \def\AMCchoiceLabel#1{%
433   \ifAMC@inside@digit\arabic{#1}%
434   \else\Alph{#1}\fi%
435 }
436 \def\AMCchoiceLabelFormat#1{#1}
437 \newcounter{AMC@ncase}
438 \setcounter{AMC@ncase}{0}
439 \newcommand\AMC@answerBox[4]{%
440   \AMC@answerBox@{\ifx\@empty#1\@empty%
441     \AMCchoiceLabel{AMC@ncase}%
442     \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.

```

443 \newlength\AMC@boxedrule
444 \newlength\AMC@crossrule
445 \newlength\AMC@boxeddown
446 \newlength\AMC@boxedwidth
447 \newlength\AMC@boxedheight
448 \newlength\AMC@oval@radius

```

```

449 \newlength\AMC@outside@sep
450 \define@choicekey{AMCdim}{shape}{square,oval,form,none}{\def\AMC@shapename{#1}}
451 \define@key{AMCdim}{size}{\AMC@boxedwidth=#1\AMC@boxedheight=#1}
452 \define@key{AMCdim}{height}{\AMC@boxedheight=#1}
453 \define@key{AMCdim}{width}{\AMC@boxedwidth=#1}
454 \define@key{AMCdim}{rule}{\AMC@boxedrule=#1}
455 \define@key{AMCdim}{outsidesep}{\AMC@outside@sep=#1}
456 \define@key{AMCdim}{down}{\AMC@boxeddown=#1}
457 \define@key{AMCdim}{color}{\def\AMC@boxcolor{#1}}
458 \define@boolkey{AMCdim}{cross}[false]{}
459 \define@key{AMCdim}{crosschar}{\textbf{\textsf{X}}}{\def\AMC@crosschar{#1}}
460 \define@key{AMCdim}{crossrule}[1.5pt]{\AMC@crossrule=#1}
461 \def\AMCboxStyle#1{%
462   \setkeys{AMCdim}{#1}%
463   \ifnum\AMC@boxedwidth<\AMC@boxedheight%
464     \AMC@oval@radius=\AMC@boxedwidth\divide\AMC@oval@radius\tw%
465   \else%
466     \AMC@oval@radius=\AMC@boxedheight\divide\AMC@oval@radius\tw%
467   \fi%
468   \csname AMC@shapeprepare@\AMC@shapename@ \endcsname%
469 }
470 \AMCboxStyle{shape=square,size=2.5ex,down=.4ex,rule=.5pt,outsidesep=.1em,color=black,cross,crosschar,crossrule}
471 \newcommand\AMCboxColor[1]{\AMCboxStyle{color=#1}}
472 \let\AMCboxDimensions=\AMCboxStyle

```

**\AMCboxOutsideLetter** Command `\AMC@box{<char>}{<answer>}` prints a box with character `<char>` inside, showing answer `<answer>` (`\AMC@checkedbox` to get a filled box), using global variables to identify the box (question and choice).

**\AMC@formBox@** It calls `\AMC@formBox@{<char>}{<answer>}{<trace>}{<key>}` to actually render the box.

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

**\AMC@formBox@** The command `\AMCboxOutsideLetter{<box>}{<char>}` is called to print the box *and* the character `<char>` 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 `\AMCformBox@` when the text that corresponds to the answer is the letter/character inside the box itself (see `\AMCcodeGrid` and `\AMCnumericChoices`).

```

473 \def\AMCoutsideLabelFormat#1{#1}
474 \newcommand\AMCboxOutsideLetter[2]{#1\nobreak\hspace{.1em}\AMCoutsideLabelFormat{#2}}
475 \newif\ifAMC@printformoutside%
476 \newcommand\ifAMC@printformoutside{%
477   \AMC@printformoutside@false%
478   \ifAMC@ensemble\ifAMC@outside@box%
479     \ifAMC@formulaire@dedans\AMC@printformoutside@true\fi%
480     \ifAMC@zoneformulaire\AMC@printformoutside@true\fi%
481   \fi\fi%
482   \ifAMC@printformoutside%
483 }
484 \newcommand\AMC@formBox@[4]{%
485   \ifAMC@printformoutside% letter to be written outside the box
486     \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%

```

```

487 \else%
488 \AMC@answerBox@{#1}{#2}{#3}{#4}%
489 \fi%
490 \AMC@tracechar{#1}{#2}{#3}{#4}%
491 }
492 \newif\ifAMC@printkeyoutside%
493 \newcommand\ifAMC@printkeyoutside{%
494 \AMC@printkeyoutside@false%
495 \ifAMC@ensemble%
496 \ifAMC@outside@box\AMC@printkeyoutside@true\fi%
497 \else%
498 \ifAMC@inside@box\else\AMC@printkeyoutside@true\fi%
499 \fi%
500 \ifAMC@printkeyoutside@%
501 }
502 \newcommand\AMC@keyBox@[4]{%
503 \ifAMC@printkeyoutside%
504 \AMC@boxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
505 \else%
506 \AMC@answerBox@{#1}{#2}{#3}{#4}%
507 \fi%
508 \AMC@tracechar{#1}{#2}{#3}{#4}%
509 }
510 \newcommand\AMC@formBox[4]{%
511 \AMC@formBox@{\ifx\empty#1\empty%
512 \AMC@choiceLabel{\AMC@ncase}%
513 \else #1\fi}{#2}{#3}{#4}%
514 }
515 \newcommand{\AMC@box}[2]{%
516 \ifAMC@ensemble%
517 \ifAMC@zoneformulaire% for codes inside form sheet
518 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
519 \else%
520 \ifAMC@formulaire@dedans% for answer boxes inside form sheet
521 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
522 \else% outside form sheet: not to be read during data capture
523 \AMC@formBox{#1}{#2}{1}{casequestion:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
524 \fi\fi%
525 \else% no separate sheet for answers: always read
526 \ifAMC@inside@box%
527 \AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
528 \else%
529 \AMC@formBox@{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
530 \fi%
531 \fi%
532 }

```

### 4.8.3 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.

```

533 \newif\ifAMCsz@logged\AMCsz@loggedfalse
534 \newcommand{\AMCscoreZone}[1]{%
535   \ifAMC@ensemble%
536     \ifAMCformulaire@dedans%
537       \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
538     \else%
539       \AMC@tracebox{1}{scorequestion::\the\AMCid@quest,-1}{#1}%
540     \fi%
541   \else%
542     \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
543   \fi%
544   \ifAMCsz@logged\else%
545     \AMC@amclog{AUTOQCM[VAR:scorezones=1]^^J}%
546     \global\AMCsz@loggedtrue%
547   \fi%
548 }
```

### 4.8.4 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`. 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@premierecopie`.  
`\AMC@NCBcheck` The length of zone reserved for binary boxes is `\AMC@CBtaille`.

```

549 \def\AMCid@checkmax{60}
550 \def\AMC@NCBetud{12}
551 \def\AMC@NCBpage{6}
552 \def\AMC@NCBcheck{6}
553 \newlength{\AMC@CBtaille}\setlength{\AMC@CBtaille}{5cm}
554 \def\AMC@premierecopie{1}
```

`\AMC@binaryBoxes` Command `\AMC@binaryBoxes[<ndigits>]{<n>}` prints `<ndigits>` boxes to represent number `<n>` in its binary form. `\AMCbin@one` and `\AMCbin@zero` print individual digit-boxes.

For example, `\AMC@binaryBoxes[12]{367}` shows  $367 = 000101101111_2$  using 12 boxes:



```

555 \newtoks\AMCbin@sequence
556 \newcount\AMCbin@number
557 \newcount\AMCbin@ndigits
558 \newcount\AMCbin@id
559 \newcount\AMCbin@digit
560 \def\AMCbin@one{\advance\AMCbin@digit\@ne%
561   \AMC@answerBox@{\AMC@checkedbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@digit}}
```



```

562 \def\AMCbin@zero{\advance\AMCbin@digit\@ne%
563   \AMC@answerBox@{}{}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@digit}}
564 \def\AMCbin@begin#1{\AMCbin@id=#1\AMCbin@digit=\z@}
565 \newcommand{\AMC@binaryBoxes}[2][1]{%
566   {\AMCboxDimensions{shape=square,size=.32cm,down=0pt,rule=.2pt,cross=false}\AMCbin@sequence={}\AMCbin@number=}
567   \AMCbin@endigits=\z@%
568   \loop%
569   \ifnum\AMCbin@number>\z@%
570   \advance\AMCbin@endigits\@ne%
571   \ifodd\AMCbin@number\AMCbin@sequence=\expandafter{\expandafter\AMCbin@one\the\AMCbin@sequence}%
572   \else\AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\fi%
573   \divide\AMCbin@number\tw@%
574   \repeat%
575   \loop\relax%
576   \ifnum\AMCbin@endigits<#1\advance\AMCbin@endigits\@ne%
577   \AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\repeat%
578   \the\AMCbin@sequence%
579   \ifnum\AMCbin@endigits>#1\PackageError{automultiplechoice}{Too low AMC@NCB value (got #1 but needs \the\AMCbin@
580   }}

```

## 4.9 Checking Environment

`\AMCcurrentenv` Sets the current environment as document.

```
581 \def\AMCcurrentenv{document}
```

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

```

582 \def\AMCif@env#1{
583   \def\AMC@tempenv{#1}%
584   \ifx\AMC@tempenv\AMCcurrentenv
585     \expandafter\@firstoftwo
586   \else
587     \expandafter\@secondoftwo
588   \fi
589 }

```

## 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.

```

590 \newcount\AMCtok@k
591 \newcount\AMCtok@max
592 \newcount\AMCtok@size
593 \newcommand{\nouveaugroupe}[2]{%

```

```

594 \expandafter\ifx\csname #1@k\endcsname\relax%
595   \expandafter\newcount\csname #1@k\endcsname%
596   \expandafter\newcount\csname AMC#1@j\endcsname%
597   \csname #1@k\endcsname=\z@\relax%
598   \csname AMC#1@j\endcsname=\z@\relax%
599   \setgroupmode{#1}{\AMCdefault@groupmode}%
600 \fi%
601 }
602 \newcommand\AMC@prepare@element[1]{%
603   \nouveau groupe{#1}{}%
604   \global\advance\csname #1@k\endcsname\@ne\relax%
605   \AMCtok@k=\csname #1@k\endcsname%
606   \expandafter\ifx\csname #1@romannumeral\AMCtok@k\endcsname\relax%
607     \expandafter\newtoks\csname #1@romannumeral\AMCtok@k\endcsname\fi%
608 }
609 \newcommand{\element}[2]{%
610   \AMC@prepare@element{#1}%
611   \csname #1@romannumeral\AMCtok@k\endcsname={#2}%
612 }

```

`\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.

```

613 \def\AMCdefault@groupmode{fixed}
614 \newcommand{\setdefaultgroupmode}[1]{\def\AMCdefault@groupmode{#1}}
615 \newcommand{\setgroupmode}[2]{%
616   \expandafter\ifx\csname AMCgrouppre@#2\endcsname\relax%
617     \PackageError{automultiplechoise}{Unknown group mode for #1 : #2}%
618     {You asked to set group '#1' mode to '#2',
619      but '#2' is not a valid group mode}%
620   \else%
621     \expandafter\global\expandafter\def\csname AMC#1@mode\endcsname{#2}%
622   \fi%
623 }

```

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  $\langle i \rangle$ , or 0 if  $\langle i \rangle$  is negative (take elements from the beginning).

```
624 \newcommand{\AMCgrouppre@fixed}[3]{%
625   \ifnum#3<\z@%
626     \csname AMC#1@j\endcsname=\z@%
627   \else%
628     \csname AMC#1@j\endcsname=#3%
629   \fi%
630 }
```

For mode **withreplacement**, the group is shuffled and the group index is set to  $\langle i \rangle$  or 0 (take elements from the beginning) if negative.

```
631 \newcommand{\AMCgrouppre@withreplacement}[3]{%
632   \ifnum#3<\z@%
633     \csname AMC#1@j\endcsname=\z@%
634   \else%
635     \csname AMC#1@j\endcsname=#3%
636   \fi%
637   \shufflegroup{#1}%
638 }
```

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.

```
639 \newcount\AMC@imax
640 \newcommand{\AMCgrouppre@withoutreplacement}[3]{%
641   \ifnum#3<\z@%
642   \else%
643     \csname AMC#1@j\endcsname=#3%
644   \fi%
645   \ifnum\AMCtok@ik=\AMCloop@k%
646     \AMCtok@ik=\z@%
647   \fi%
648   \ifnum\AMCtok@ik=\z@%
649     \shufflegroup{#1}%
650   \else%
651     \AMC@imax=\AMCloop@k%
652     \advance\AMC@imax -#2\relax%
653     \ifnum\AMCtok@ik>\AMC@imax%
654       \shufflegroupslice{#1}{\@ne}{\AMCtok@ik}%
655     \ifnum\AMCtok@ik<\AMCloop@k%
656       \advance\AMCtok@ik\@ne%
657       \shufflegroupslice{#1}{\AMCtok@ik}{\AMCloop@k}%
658     \fi%
659   \fi%
660 \fi%
661 }
```

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

```
662 \newcommand{\AMCgrouppre@cyclic}[3]{%
663   \ifnum#3<\z@%
```

```

664 \else%
665 \csname AMC#1@j\endcsname=#3%
666 \fi%
667 }

```

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

```

668 \newcommand{\AMCgroup@pre}[4]{%
669 \csname AMCgroup@pre@#1\endcsname{#2}{#3}{#4}%
670 }

```

`\shufflegroup` Command `\shufflegroup{<group-name>}` shuffles the elements of group `<group-name>`, and  
`\insertgroup` `\shufflegroupslice{<group-name>}{<a>}{<b>}` shuffles elements `<a>` to `<b>` from group `<group-name>`.  
`\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 `<groupname>`, or only the first `<n>` elements if `<n>` is given. `\insertgroupfrom[<n>]{<groupname>}{<i>}` inserts all the elements of group `<groupname>` starting from index `<i>` (the index of the first element is 0), or only the first `<n>` elements if `<n>` is given.

```

671 \newcommand{\shufflegroup}[1]{%
672 \ifAMC@shuffleG{\AMC@shuffletoks{number\csname #1@k\endcsname}{#1@}}\fi%
673 }
674 \newcommand{\shufflegroupslice}[3]{%
675 \ifAMC@shuffleG{\AMC@shuffletoks[#2]{#3}{#1@}}\fi%
676 }
677 \newcount\AMCtok@ik
678 \newcount\AMCloop@k
679 \newcommand{\AMCgrouploop@prep}[3]{%
680 \AMCtok@size=#1\relax%
681 \ifAMC@fullGroups\AMCtok@size=\z@\fi%
682 \ifnum\AMCtok@size<\@ne%
683 \AMCtok@size=\csname #2@k\endcsname%
684 \fi%
685 \AMCtok@ik=\csname AMC#2@j\endcsname%
686 \AMCloop@k=\csname #2@k\endcsname%
687 \expandafter\ifx\csname AMC#2@mode\endcsname\relax%
688 \PackageError{automultiplechoice}{No group mode for #2}%
689 {No mode has been defined for group ' #2'. This should not occur...}%
690 \fi%
691 \AMCgroup@pre{\csname AMC#2@mode\endcsname}{#2}{\the\AMCtok@size}{#3}%
692 }
693 \newcommand{\AMCgrouploop@next}[1]{%
694 \global\advance\csname AMC#1@j\endcsname\@ne\relax%
695 \expandafter\ifnum\csname AMC#1@j\endcsname>\AMCloop@k\relax%
696 \global\csname AMC#1@j\endcsname=\@ne%
697 \fi%
698 \AMCtok@ik=\csname AMC#1@j\endcsname%
699 \advance\AMCtok@size\m@ne%
700 }
701 \newcommand{\insertgroupfrom}[3][0]{%
702 \AMCgrouploop@prep{#1}{#2}{#3}%

```

```

703 {\loop%
704   \AMCgrouploop@next{#2}%
705   {\the\csname #2@\romannumeral\AMCtok@ik\endcsname}%
706   \ifnum\AMCtok@size>\z@\repeat}%
707 }
708 \newcommand{\insertgroup}[2][0]{%
709   \insertgroupfrom[#1]{#2}{-1}%
710 }

```

`\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.

```

711 \newcommand{\cleargroup}[1]{%
712   \nouveaugroupe{#1}{}%
713   \csname #1@k\endcsname=\z@\relax%
714   \csname AMC#1@j\endcsname=\z@\relax%
715 }
716 \newcommand{\copygroupfrom}[4][0]{%
717   \AMCgrouploop@prep{#1}{#2}{#4}%
718   {\loop%
719     \AMCgrouploop@next{#2}%
720     \AMC@prepare@element{#3}%
721     \global\csname #3@\romannumeral\AMCtok@k\endcsname=\csname #2@\romannumeral\AMCtok@ik\endcsname%
722     \ifnum\AMCtok@size>\z@\repeat}%
723 }
724 \newcommand{\copygroup}[3][0]{%
725   \copygroupfrom[#1]{#2}{#3}{-1}%
726 }

```

## 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`.

```

727 \newcount\AMCrep@count
728 \AMCload@counter=199
729 \@whilenum\AMCload@counter>0\do{%
730   \expandafter\newtoks\csname reponse@\romannumeral\AMCload@counter\endcsname%
731   \advance\AMCload@counter\m@ne%
732 }

```

`\AMCload@reponse` Command `\AMCload@reponse{<n>}{<text>}` will be used to add answer number  $\langle n \rangle$  with text  $\langle text \rangle$  ( $\langle text \rangle$  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  $\langle text \rangle$ .

```
733 \newcommand\AMCload@reponse[2]{%
734   \advance\AMCload@counter\@ne\relax%
735   \csname reponse@\romannumeral\AMCload@counter\endcsname%
736   =\expandafter{\expandafter\AMCrep@count\expandafter=#2 #1}%
737 }
738 \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.

```
739 \def\shuffle@it{\AMC@shuffletoks{\number\AMCload@counter}{reponse@}}
740 \newcount\AMCnum@questions
741 \newcommand\AMCdump@reponses{%
742   \global\AMCnum@questions=\AMCload@counter%
743   \@whilenum\AMCload@counter>0\do{%
744     \the\csname reponse@\romannumeral\AMCload@counter\endcsname%
745     \advance\AMCload@counter\m@ne}}
```

#### 4.11.1 Managing answers

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

- If  $\langle mode \rangle = r$ , `\AMCload@@reponse` is `\AMCload@reponse` (loads answer to token register) and `\AMCrep@fini` calls `\shuffle@it` and `\AMCdump@reponses`;
- If  $\langle mode \rangle = o$ , `\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`.

```
746 \newcommand\AMCrep@init[1]{%
747   \ifAMC@ordre\AMCrep@o\else%
748     \csname AMCrep@#1\endcsname\fi\AMCload@counter=\z@}
749 \newcommand\AMCrep@o{%
750   \def\AMCload@@reponse{\AMCrien@deux}\def\AMCrep@fini{}}
```

```

751 \newcommand\AMCrep@r{%
752   \def\AMCload@reponse{\AMCload@reponse}%
753   \def\AMCrep@fini{\shuffle@it\AMCdump@reponses}}
754 \newcount\AMCrep@count
755 \newcommand\lastchoices{%
756   \AMCrep@count=\AMCrep@count%
757   \AMCrep@fini\AMCrep@init{o}%
758   \AMCrep@count=\AMCrep@count}
759 \newcommand\@aucune{\emph{\AMC@loc@none}}
760 \newcommand\AMC@fin@rep{%
761   \ifAMCcomplete@multi\ifAMCtype@multi%
762     \lastchoices\AMCrep@count=-1%
763     \ifAMC@bonne@wrongchoice{\@aucune}\else%
764       \ifAMC@postcorrect@wrongchoice{\@aucune}\else\correctchoice{\@aucune}\fi%
765     \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 (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.

```

766 \def\AMCformBeforeQuestion{\vspace{\AMCformVSpace}\par}
767 \def\AMCformAfterQuestion{\ifAMC@asqbloc\egroup\fi}
768 \def\AMCformQuestion#1{\AMC@loc@qf{#1}}
769 \def\AMCformQuestionN{\AMCformQuestion{\AMC@qaff}}
770 \def\AMCformQuestionA{%
771   \setcounter{AMC@ncase}{0}%
772   \AMCformBeforeQuestion%
773   \ifAMC@asqbloc\vbox\bgroup\fi%
774   \ifx\@empty\AMC@sza@callout\@empty\else%
775     \csname\AMC@sza@callout\endcsname%
776   \fi%
777   \AMCformQuestionN%
778   \ifx\@empty\AMC@sza@callin\@empty\else%
779     \csname\AMC@sza@callin\endcsname%
780   \fi%
781 }
782 \def\AMCformAnswer#1{\hspace{\AMCformHSpace} #1}
783 \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>}` adds `<code>` to this memory. `\AMC@mem@answer{<code>}` adds to memory answer code `<code>`, and `\AMCformS`

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

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.

```

784 \ExplSyntaxOn
785
786 \prg_set_conditional:Nnn \amc_if_separate_question: { p , T } {
787   \ifAMC@ensemble
788     \ifAMC@zoneformulaire
789       \prg_return_false:
790     \else
791       \prg_return_true:
792     \fi
793   \else
794     \prg_return_false:
795   \fi
796 }
797 \cs_new_eq:NN \AMC@if@separate@question \amc_if_separate_question:T
798
799 \int_new:N \amc_memory_elts_count
800
801 \cs_new:Nn \amc_clear_memory: { \int_gzero:N \amc_memory_elts_count }
802 \cs_new_eq:NN \AMC@mem@clear \amc_clear_memory:
803
804 \cs_new:Npn \amc_memory_elt_i:n #1 {
805   amc_memory_elts_ \int_to_alph:n { #1 }
806 }
807 \cs_new:Nn \amc_memory_current_elt: {
808   \amc_memory_elt_i:n \amc_memory_elts_count
809 }
810 \cs_new:Npn \amc_memory_vars_i:n #1 {
811   amc_memory_vars_ \int_to_alph:n { #1 }
812 }
813 \cs_new:Nn \amc_memory_current_vars: {
814   \amc_memory_vars_i:n \amc_memory_elts_count
815 }
816
817 \cs_new:Nn \amc_add_memory_elt: {
818   \int_gincr:N \amc_memory_elts_count
819   \tl_gclear_new:c { \amc_memory_current_elt: }
820   \tl_gclear_new:c { \amc_memory_current_vars: }
821 }
822 \cs_new_eq:NN \AMC@mem@next \amc_add_memory_elt:
823
824 \cs_new:Npn \amc_add_to_memory:n #1 {
825   \tl_gput_right:cn { \amc_memory_current_elt: } { #1 }
826 }

```



```

827 \cs_new_eq:NN \AMC@mem@add \amc_add_to_memory:n
828
829 \cs_new:Npn \amc_add_to_vars:n #1 {
830   \tl_gput_right:cn { \amc_memory_current_vars: } { #1 }
831 }
832 \cs_new_eq:NN \AMC@mem@addvar \amc_add_to_vars:n
833
834 \cs_new:Npn \amc_add_qidaffname:nnn #1#2#3 {
835   \amc_add_to_vars:n {\AMCid@quest=#1\setcounter{AMCquestionaff}{#2}%
836     \global\def\AMCid@name{#3}}
837 }
838 \cs_generate_variant:Nn \amc_add_qidaffname:nnn { xxx }
839 \cs_new_eq:NN \AMC@mem@qidaffname \amc_add_qidaffname:xxx
840
841 \cs_new:Npn \amc_mem_elt_cat:n #1 {
842   \amc_add_to_vars:n { \def\AMCmem@elt@cat{ #1 } }
843 }
844 \cs_generate_variant:Nn \amc_mem_elt_cat:n { x }
845 \cs_new_eq:NN \AMC@mem@category \amc_mem_elt_cat:x
846
847 \cs_new:Npn \amc_add_aid:n #1 {
848   \amc_add_to_memory:n {\AMCrep@count=#1}
849 }
850 \cs_generate_variant:Nn \amc_add_aid:n { x }
851 \cs_new_eq:NN \AMC@mem@aid \amc_add_aid:x
852
853 \cs_new:Npn \amc_if_category_is_p:n #1 {
854   \str_if_eq_p:on { \AMCmem@elt@cat } { #1 }
855 }
856 \cs_new:Npn \amc_use_memory:n #1 {
857   \int_step_inline:nnnn { 1 } { 1 } \amc_memory_elts_count {
858     \def\AMCmem@elt@cat{ plain }
859     \tl_use:c { \amc_memory_vars_i:n { ##1 } }
860     \bool_if:nTF { #1 } {
861       \tl_use:c { \amc_memory_elt_i:n { ##1 } }
862     } { }
863   }
864 }
865 \cs_new:Nn \amc_use_memory: { \amc_use_memory:n { \c_true_bool } }
866 \cs_new_eq:NN \AMC@mem@show \amc_use_memory:
867 \cs_new_eq:NN \AMC@mem@show@filter \amc_use_memory:n
868 \cs_new_eq:NN \AMC@category \amc_if_category_is_p:n
869
870 \ExplSyntaxOff
871 \newcommand\AMC@mem@add@ifneeded[1]{%
872   \AMC@if@separate@question{%
873     \AMC@mem@add{#1}%
874   }%
875 }
876 \newcommand\AMC@mem@addsingle@ifneeded[2]{%

```

```

877 \AMC@if@separate@question{%
878   \AMC@mem@next%
879   \AMC@mem@category{#2}%
880   \AMC@mem@add{#1}%
881 }%
882 }
883 \newcommand\AMC@mem@answer[1]{%
884   \AMC@if@separate@question{%
885     \AMC@mem@aid{\the\AMC@rep@count}%
886     \AMC@mem@add{\AMC@form@answerA{#1}}%
887   }%
888 }
889 \newcommand\AMC@mem@openQuestion{%
890   \AMC@if@separate@question{%
891     \AMC@mem@next%
892     \AMC@mem@qidaffname{\the\AMC@id@quest}{\arabic{AMC@questionaff}}{\AMC@id@name}%
893     \AMC@mem@add{\AMC@form@questionA}%
894   }%
895 }
896 \def\AMC@form@begin{%
897   \AMC@zone@formulairetrue\setcounter{section}{0}%
898   \ifAMC@ensemble\ifAMC@autemarks\pagestyle{AMC@pageFull}\fi\fi%
899 }
900 \newcommand\AMC@form{%
901   \ifAMC@ensemble\AMC@formulaire@dedanstrue%
902     \AMC@mem@show%
903   \fi}
904 \newcommand\AMC@form@filter[1]{%
905   \ifAMC@ensemble\AMC@formulaire@dedanstrue%
906     \AMC@mem@show@filter{#1}%
907   \fi}
908 \newif\ifAMC@keepmemory
909 \newcommand\AMC@form@s{%
910   \ifAMC@ensemble\AMC@formulaire@dedanstrue%
911     \AMC@amclog{AUTOQCM[BR=0]^^J}\AMC@mem@show%
912     \AMC@keepmemorytrue%
913   \fi}

```

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

```

914 \newcommand{\AMC@sectionNumbered}[1]{%
915   \section{#1}\AMC@mem@addsingle@ifneeded{\section{#1}}{section}}
916 \newcommand{\AMC@subsectionNumbered}[1]{%
917   \subsection{#1}\AMC@mem@addsingle@ifneeded{\subsection{#1}}{subsection}}
918 \newcommand{\AMC@sectionStar}[1]{%
919   \section*{#1}\AMC@mem@addsingle@ifneeded{\section*{#1}}{section}}
920 \newcommand{\AMC@subsectionStar}[1]{%
921   \subsection*{#1}\AMC@mem@addsingle@ifneeded{\subsection*{#1}}{subsection}}
922 \def\AMC@section{\@ifstar\AMC@sectionStar\AMC@sectionNumbered}
923 \def\AMC@subsection{\@ifstar\AMC@subsectionStar\AMC@subsectionNumbered}

```

### 4.11.3 Formatting answers

<code>choices</code>	Answers have to be included in an environment <code>choices</code> (standard), <code>choiceshoriz</code> (answers on one line) or <code>choicescustom</code> (user defined) depending on the desired formatting.
<code>choiceshoriz</code>	
<code>choicescustom</code>	Use <code>\AMCBoxedAnswers</code> 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.
<code>\AMCBoxedAnswers</code>	<pre> 924 \def\AMCBoxedAnswers{\AMC@rbloctrue} 925 \newenvironment{choices}[1][r]{% 926   \AMCrep@count=\z@\def\une@rep{\AMCrep@itemize}% 927   \ifAMC@rbloc\def\une@rep{\AMCrep@bloc}% 928   \else\begin{itemize}\setlength{\itemsep}{\AMCinterIrep}\fi% 929     \AMCrep@init{#1}}% 930   {\AMC@fin@rep\ifAMC@rbloc\else\end{itemize}\fi} 931 \newenvironment{choiceshoriz}[1][r]{% 932   \AMCrep@count=\z@\def\une@rep{\AMCrep@ligne}\AMCrep@init{#1}% 933   \par\begin{center}}% 934   {\AMC@fin@rep\end{center}} 935 \newenvironment{choicescustom}[1][r]{% 936   \AMCrep@count=\z@\def\une@rep{\AMCrep@perso}\AMCrep@init{#1}% 937   \AMCbeginAnswer\ignorespaces}% 938   {\AMC@fin@rep\AMCendAnswer} </pre>
<code>\AMCrep@bloc</code>	For each of these styles, a corresponding <code>\AMCrep@xxx{&lt;box&gt;}{&lt;text&gt;}</code> is defined, which will format the answer with a box given in <code>&lt;box&gt;</code> and text <code>&lt;text&gt;</code> . <code>\AMCrep@bloc</code> 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.
<code>\AMCrep@itemize</code>	
<code>\AMCrep@ligne</code>	
<code>\AMCrep@perso</code>	
	<pre> 939 \newcommand\AMCrep@bloc[2]{\AMC@mem@answer{#1}% 940   \par\noindent\begin{minipage}{\linewidth}% 941     \begin{itemize}\item{#1} #2\end{itemize}\end{minipage}% 942     \vspace{\AMCinterBrep}} 943 \newcommand\AMCrep@itemize[2]{\AMC@mem@answer{#1}\item{#1} #2} 944 \newlength\AMChorizAnswerSep 945 \setlength{\AMChorizAnswerSep}{3em plus 4em} 946 \newlength\AMChorizBoxSep 947 \setlength{\AMChorizBoxSep}{1em} 948 \newcommand\AMCrep@ligne[2]{\AMC@mem@answer{#1}% 949   \mbox{#1\hspace*{\AMChorizBoxSep}#2}\hspace{\AMChorizAnswerSep}} 950 \newcommand\AMCrep@perso[2]{\AMC@mem@answer{#1}\AMCanswer{#1}{#2}} </pre>
<code>\AMCbeginAnswer</code>	The custom style will use user-defined commands to format answers: <code>\AMCbeginAnswer</code> is called once before answers, <code>\AMCanswer{&lt;box&gt;}{&lt;text&gt;}</code> is called for each answer ( <code>&lt;box&gt;</code> being the box to be ticked and <code>&lt;text&gt;</code> the text associated with the proposed answer), and <code>\AMCendAnswer</code> is called after all answers.
<code>\AMCendAnswer</code>	
<code>\AMCanswer</code>	
	<pre> 951 \def\AMCbeginAnswer{} 952 \def\AMCanswer#1#2{#1 #2} 953 \def\AMCendAnswer{} </pre>
<code>\correctchoice</code>	The commands <code>\correctchoice</code> and <code>\wrongchoice</code> are used inside <code>choices</code> -like environments to give the proposed answers and specify if they are to be ticked by the students or not.
<code>\wrongchoice</code>	
	<pre> 954 \newcommand{\correctchoice}[2][\global\advance\AMCrep@count\@ne\relax% </pre>

```

955 \ifAMC@calibration\AMC@amclog{AUTOQCM[REP=\the\AMCrep@count:B]^^J}\fi%
956 \global\AMCune@bonnettrue%
957 \AMCload@@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkedbox}%
958 \else\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}\ignorespaces}
959 \newcommand{\wrongchoice}[2][\global\advance\AMCrep@count\@ne\relax%
960 \ifAMC@calibration\AMC@amclog{AUTOQCM[REP=\the\AMCrep@count:M]^^J}\fi%
961 \AMCload@@reponse{\une@rep{\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}%
962 \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 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).

```

963 \newcommand{\AMCemptybox}[3]{\%
964 \sbox0{\wd0=#1\ht0=#2\dp0=#3\relax\box0}}
965 \newlength\AMC@mn@test
966 \newlength\AMC@mn@sep\AMC@mn@sep=4mm
967 \newlength\AMC@mn@leftmargin
968 \newlength\AMC@mn@rightmargin
969 \newcommand\AMCmarginNote[1]{\%
970 \begin{tikzpicture}[remember picture,overlay]\%
971 \coordinate (here) at (0,0);\%
972 \pgfextractx{\AMC@mn@test}{\pgfpointdiff{\pgfpointorigin}%
973 {\pgfpointanchor{current page}{center}}}\%
974 \ifodd\thepage\%
975 \AMC@mn@leftmargin=\oddsidemargin\%
976 \AMC@mn@rightmargin=\evensidemargin\%
977 \else
978 \AMC@mn@leftmargin=\evensidemargin\%
979 \AMC@mn@rightmargin=\oddsidemargin\%
980 \fi
981 \ifdim\AMC@mn@test < 1cm\%
982 \draw (current page.east |- here)+(-\AMC@mn@rightmargin-1in+\AMC@mn@sep,0pt) node[anchor=text,align=left]{#1}\%
983 \else\%
984 \draw (current page.west |- here)+(0cm,0pt) node[anchor=text,align=right,text width=\AMC@mn@leftmargin]{#1}\%
985 \fi\%
986 \end{tikzpicture}}\%
987 }

```

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)

```

988 \newcommand{\AMC@sz@box}{\AMCemptybox{\AMC@sz@width}{\AMC@sz@height}{\AMC@sz@depth}}
989 %
990 \newcommand{\AMC@sz@callin@question}{\AMCscoreZone{\AMC@sz@box}}
991 %
992 \newcommand{\AMC@sz@callout@margin}{\hspace{Opt}\marginpar{\AMCscoreZone{\AMC@sz@box}}}
993 %
994 \newcommand{\AMC@sz@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper margin}}
995 \newcommand{\AMC@sz@callout@margins}{\hspace{Opt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}

```

Let us now set up options handling.

```

996 \newlength\AMC@sz@width
997 \newlength\AMC@sz@height
998 \newlength\AMC@sz@depth
999 \def\AMC@sz@callout{}
1000 \def\AMC@sz@callin{}
1001 \define@key{AMCsz}{width}{\AMC@sz@width=#1}
1002 \define@key{AMCsz}{height}{\AMC@sz@height=#1}
1003 \define@key{AMCsz}{depth}{\AMC@sz@depth=#1}
1004 \define@key{AMCsz}{calloutsideside}{\def\AMC@sz@callout{#1}}
1005 \define@key{AMCsz}{callinsideside}{\def\AMC@sz@callin{#1}}
1006 \define@choicekey{AMCsz}{position}{none,question,margin,margins}{%
1007   \ifcsname AMC@sz@callout@#1\endcsname%
1008     \def\AMC@sz@callout{\AMC@sz@callout@#1}%
1009   \else%
1010     \def\AMC@sz@callout{}%
1011   \fi%
1012   \ifcsname AMC@sz@callin@#1\endcsname%
1013     \def\AMC@sz@callin{\AMC@sz@callin@#1}%
1014   \else%
1015     \def\AMC@sz@callin{}%
1016   \fi%
1017   \ifcsname AMC@sz@init@#1\endcsname%
1018     \csname AMC@sz@init@#1\endcsname%
1019   \fi%
1020 }
1021 \newcommand{\AMCsetScoreZone}[1]{\setkeys{AMCsz}{#1}}
1022 \AMCsetScoreZone{width=1.5em,height=1.5ex,depth=.5ex,position=none}

```

And do the same for \AMCsetScoreZoneAnswerSheet...

```

1023 \newcommand{\AMC@sza@box}{\AMCemptybox{\AMC@sza@width}{\AMC@sza@height}{\AMC@sza@depth}}
1024 %
1025 \newcommand{\AMC@sza@init@none}{}
1026 \newcommand{\AMC@sza@callout@none}{}
1027 \newcommand{\AMC@sza@callin@none}{}
1028 %
1029 \newcommand{\AMC@sza@init@question}{}
1030 \newcommand{\AMC@sza@callout@question}{}
1031 \newcommand{\AMC@sza@callin@question}{\AMCscoreZone{\AMC@sza@box}}
1032 %

```

```

1033 \newcommand{\AMC@sza@init@margin}{}
1034 \newcommand{\AMC@sza@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sza@box}}}
1035 \newcommand{\AMC@sza@callin@margin}{}
1036 %
1037 \newcommand{\AMC@sza@init@margin}{\PackageWarning{automultiplechoice}{Please run twice to get proper margin}}
1038 \newcommand{\AMC@sza@callout@margin}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}
1039 \newcommand{\AMC@sza@callin@margin}{}
1040 %
1041 \newlength\AMC@sza@width
1042 \newlength\AMC@sza@height
1043 \newlength\AMC@sza@depth
1044 \def\AMC@sza@callout{}
1045 \def\AMC@sza@callin{}
1046 \define@key{AMC@sza}{width}{\AMC@sza@width=#1}
1047 \define@key{AMC@sza}{height}{\AMC@sza@height=#1}
1048 \define@key{AMC@sza}{depth}{\AMC@sza@depth=#1}
1049 \define@key{AMC@sza}{calloutsides}{\def\AMC@sza@callout{#1}}
1050 \define@key{AMC@sza}{callinsides}{\def\AMC@sza@callin{#1}}
1051 \define@choicekey{AMC@sza}{position}{none,question,margin,margins}{%
1052   \ifcsname AMC@sza@callout@#1\endcsname%
1053     \def\AMC@sza@callout{AMC@sza@callout@#1}%
1054   \else%
1055     \def\AMC@sza@callout{}%
1056   \fi%
1057   \ifcsname AMC@sza@callin@#1\endcsname%
1058     \def\AMC@sza@callin{AMC@sza@callin@#1}%
1059   \else%
1060     \def\AMC@sza@callin{}%
1061   \fi%
1062   \ifcsname AMC@sza@init@#1\endcsname%
1063     \csname AMC@sza@init@#1\endcsname%
1064   \fi%
1065 }
1066 \newcommand{\AMCsetScoreZoneAnswerSheet}[1]{\setkeys{AMC@sza}{#1}}
1067 \AMCsetScoreZoneAnswerSheet{width=1.5em,height=1.5ex,depth=.5ex,position=none}
1068 \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` `\AMC@qaff` `\AMC@qaff` will increase this counter and `\AMC@qaff` will format the question number out.

```

1069 \newcounter{AMCquestionaff}
1070 \newcommand{\AMCnumero}[1]{\setcounter{AMCquestionaff}{#1}\addtocounter{AMCquestionaff}{-1}}
1071 \AtBeginDocument{%
1072   \ifx\@skiphyperreftrue\undefined%
1073     \expandafter\newif\csname if@skiphyperref\endcsname%
1074   \fi%

```

```

1075 }
1076 \newcommand\AMC@stepQuestion{\ifAMC@questionNumber\@skiphyperreftrue\refstepcounter{AMC@questionaff}\@skiphyper
1077 \newcommand\AMC@qaff{\arabic{AMC@questionaff}}

```

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

```

1078 \def\AMCbeforeQuestion{\ifAMC@qbloc\else\par\noindent\fi}
1079 \def\AMCbeginQuestion#1#2{\noindent\AMC@loc@q{#1}{#2}%
1080 \ifx\@empty\AMC@sz@callin\@empty\hspace*{1em}\fi%
1081 }
1082 \def\multiSymbole{${\clubsuit}$}

```

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

`questionouverte` Environment `{questionmult}{<key>}` is the same for multiple questions (with none, one or  
`\ouverte@vs` several correct choices).

Environment `{questionmultx}{<key>}` is the same as `questionmult`, but with no use of  
`\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).

```

1083 \ifx\question\undefined\else\let\question\undefined\fi
1084 \def\AMCnobloc{\AMC@qblocfalse}
1085 \def\AMCbloc{\AMC@qbloctrue}
1086 \newenvironment{question}[2][{}]{%
1087 \def\AMCcurrentenv{question}%
1088 \AMC@stepQuestion%
1089 \global\def\AMCid@name{#2}\AMC@affiche{#2}{\AMCid@quest}%
1090 \ifAMC@calibration\AMCmessage{Q=\the\AMCid@quest}\fi%
1091 \AMCbeforeQuestion%
1092 \ifx\@empty\AMC@sz@callout\@empty\else%
1093 \csname\AMC@sz@callout\endcsname%
1094 \fi%
1095 \AMCtype@multifalse\ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1096 \ifAMC@affichekeys\index{\texttt{#2}}\fi%
1097 \AMCbeginQuestion{\ifAMC@affichekeys\ifAMC@ensemble\AMC@qaff\ \fi[\texttt{#2}]\else\AMC@qaff\fi}{#1}%
1098 \ifx\@empty\AMC@sz@callin\@empty\else%
1099 \csname\AMC@sz@callin\endcsname%
1100 \fi%
1101 \AMCformulaire@dedansfalse\setcounter{AMC@ncase}{0}%
1102 \AMC@mem@openQuestion}%
1103 {\ifAMC@qbloc\end{minipage}\vspace{\AMCinterBquest}\else\vspace{\AMCinterIquest}\fi\AMCmessage{FQ}\AMC@mem@a
1104 \newenvironment{questionmult}[1]{%
1105 \AMC@bonnefalse\begin{question}[\multiSymbole]{#1}%
1106 \AMCtype@multittrue\ifAMC@calibration%
1107 \AMC@amclog{AUTOQCM[MULT]^^J}\fi%
1108 {\end{question}}
1109 \newenvironment{questionmultx}[1]{%

```

```

1110 \begingroup\def\multiSymbole{}\begin{questionmult}{#1}}%
1111 {\end{questionmult}\endgroup}
1112 \newdimen\ouverte@vs
1113 \newenvironment{questionouverte}[1][3cm]{%
1114 \AMC@stepQuestion%
1115 \AMCtype@multifalse\ouverte@vs=#1%
1116 \ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1117 \AMCbeginQuestion{\AMC@qaff}{}}%
1118 {\vspace*{\ouverte@vs}\ifAMC@qbloc\end{minipage}\vspace{3ex}\fi}

```

#### 4.11.6 Explanations

`\explain` The command `\explain` is used inside question-like environments to give the explanation for the answers of a question.

```

1119 \newcommand{\explain}[1]{%
1120 \ifAMC@correthead%
1121 \AMCif@env{question}{\par\noindent{\AMC@loc@explain #1}}{\AMC@error@explain}\vspace{1ex}%
1122 \else%
1123 \AMCif@env{question}{}{\AMC@error@explain}%
1124 \fi%
1125 }

```

#### 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  
`QuestionIndicative` simple and multiple questions, and `\QuestionIndicative` tells that the current question is not  
no be taken into account in the global mark.

```

1126 \def\scoring#1{\ifAMC@calibration\AMC@amclog{AUTOQCM[B=#1]^^J}\fi}
1127 \def\scoringDefaultS#1{\ifAMC@calibration\AMC@amclog{AUTOQCM[BDS=#1]^^J}\fi}
1128 \def\scoringDefaultM#1{\ifAMC@calibration\AMC@amclog{AUTOQCM[BDM=#1]^^J}\fi}
1129 \def\QuestionIndicative{\ifAMC@calibration\AMC@amclog{AUTOQCM[INDIC]^^J}\fi}

```



## 4.13 Numerical data

### 4.13.1 Codes

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], \dots, \langle key \rangle[\langle length(descr) \rangle]`.

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"/>	1	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1
<input type="text"/>	A	<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"/>	C	<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"/>	0	<input type="text"/>	0	<input type="text"/>	0	<input type="text"/>	0
<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1	<input type="text"/>	1
<input type="text"/>	A	<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"/>	C	<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

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.

```

1130 \newcount\AMC@chiffres
1131 \newdimen\AMCcodeHspace\AMCcodeHspace=.5em
1132 \newdimen\AMCcodeVspace\AMCcodeVspace=.5em
1133 \ExplSyntaxOn
1134
1135 \clist_new:N \amc_code_descr_clist
1136 \seq_new:N \amc_code_digits_seq
1137 \int_new:N \amc_code_digit_n_int
1138 \bool_new:N \amc_code_vertical_bool

```

```

1139 \bool_new:N \amc_code_top_bool
1140
1141 \cs_new:Npn \amc_code_init:N #1 {
1142   \def\AMCbeginQuestion##1##2{}
1143   \def\AMCbeforeQuestion{}
1144   \AMCnoScoreZone
1145   \AMCquestionNumberfalse
1146   \setlength{\parindent}{0pt}
1147   \AMCnobloc
1148   \int_set:Nn \amc_code_digit_n_int { \clist_count:N #1 }
1149 }
1150
1151 \cs_new:Nn \amc_code_digit_init: {
1152   \QuestionIndicative
1153   \global\AMCrep@count=\z@
1154 }
1155
1156 \cs_new:Npn \amc_code_digit:n #1 {
1157   \global\advance\AMCrep@count\@ne\relax
1158   \ifAMC@calibration\AMC@amclog{AUTOQCM[ REP = \the\AMCrep@count : M ]^^J}\fi
1159   \hbox{\AMC@keyBox@{#1}}{1}{case : \AMCid@name : \the\AMCid@quest , \the\AMCrep@count}}
1160   \bool_if:NTF \amc_code_vertical_bool {
1161     \vspace{\AMCcodeVspace}
1162   }{
1163     \hspace{\AMCcodeHspace}
1164   }
1165 }
1166
1167 \keys_define:nn { amccode } {
1168   vertical .bool_set:N = \amc_code_vertical_bool,
1169   vertical .initial:n = { true },
1170   vertical .default:n = { true },
1171   v .code:n = { \bool_set_true:N \amc_code_vertical_bool },
1172   h .code:n = { \bool_set_false:N \amc_code_vertical_bool },
1173   top .bool_set:N = \amc_code_top_bool,
1174   top .initial:n = { false },
1175   top .default:n = { true }
1176 }
1177
1178 \cs_new:Npn \amc_code_generate:nNn #1#2#3 {
1179   { \keys_set:nn { amccode } { #3 }
1180     \amc_code_init:N #2
1181     \clist_map_inline:Nn #2 { % iterates over 'digits'
1182       \begin{question}{#1[ \int_use:N \amc_code_digit_n_int ]}
1183         \amc_code_digit_init:
1184         \seq_set_split:Nnn \amc_code_digits_seq {} { ##1 }
1185         \bool_if:NTF \amc_code_vertical_bool {
1186           \hspace{0pt}
1187         } \bool_if:NTF \amc_code_top_bool { \vtop } { \vbox }
1188         \bgroup

```

```

1189     }{
1190         \hbox\bgroup
1191     }
1192     \seq_map_inline:Nn \amc_code_digits_seq {
1193         % iterates over available characters for 'digit'
1194         \amc_code_digit:n { ####1 }
1195     }
1196     \bool_if:NTF \amc_code_vertical_bool {
1197         \vspace{-\AMCcodeVspace}\egroup
1198         \hspace{\AMCcodeHspace}
1199     }{
1200         \egroup\vspace{\AMCcodeVspace}
1201         \par
1202     }
1203     \end{question}
1204     \int_decr:N \amc_code_digit_n_int
1205 }
1206 }
1207 }
1208
1209 \cs_new:Npn \amc_code_generate:nnn #1#2#3 {
1210     \clist_set:Nn \amc_code_descr_clist { #2 }
1211     \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1212 }
1213 \cs_generate_variant:Nn \amc_code_generate:nnn { xxx }
1214 \newcommand{\AMCcodeGrid}[3][[]]{
1215     \amc_code_generate:xxx { #2 } { #3 } { #1 }
1216 }
1217
1218 \cs_new:Npn \amc_code_generate_integer:nnn #1#2#3 {
1219     \clist_clear:N \amc_code_descr_clist
1220     \prg_replicate:nn { #2 } { \clist_put_right:Nn \amc_code_descr_clist { 0123456789 } }
1221     \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1222 }
1223 \cs_generate_variant:Nn \amc_code_generate_integer:nnn { xxx }
1224 \newcommand{\AMCcodeGridInt}[3][[]]{
1225     \amc_code_generate_integer:xxx { #2 } { #3 } { #1 }
1226 }
1227
1228 \cs_new:Npn \amc_code_generate_integer_v:nn #1#2 {
1229     \amc_code_generate_integer:nnn { #1 } { #2 } { v }
1230 }
1231 \cs_new:Npn \amc_code_generate_integer_h:nn #1#2 {
1232     \amc_code_generate_integer:nnn { #1 } { #2 } { h }
1233 }
1234 \cs_generate_variant:Nn \amc_code_generate_integer_v:nn { xx }
1235 \cs_generate_variant:Nn \amc_code_generate_integer_h:nn { xx }
1236 \cs_new_eq:NN \AMCcode \amc_code_generate_integer_v:xx
1237 \cs_new_eq:NN \AMCcodeH \amc_code_generate_integer_h:xx
1238

```

### 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 3</b> What is the value of $7 \times 5$ ?																				
<div style="border: 1px solid black; display: inline-block; padding: 5px;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">0</td> <td style="border: 1px solid black; width: 20px; text-align: center;">1</td> <td style="border: 1px solid black; width: 20px; text-align: center;">2</td> <td style="border: 1px solid black; width: 20px; text-align: center;">3</td> <td style="border: 1px solid black; width: 20px; text-align: center;">4</td> <td style="border: 1px solid black; width: 20px; text-align: center;">5</td> <td style="border: 1px solid black; width: 20px; text-align: center;">6</td> <td style="border: 1px solid black; width: 20px; text-align: center;">7</td> <td style="border: 1px solid black; width: 20px; text-align: center;">8</td> <td style="border: 1px solid black; width: 20px; text-align: center;">9</td> </tr> <tr> <td style="border: 1px solid black; width: 20px; text-align: center;">0</td> <td style="border: 1px solid black; width: 20px; text-align: center;">1</td> <td style="border: 1px solid black; width: 20px; text-align: center;">2</td> <td style="border: 1px solid black; width: 20px; text-align: center;">3</td> <td style="border: 1px solid black; width: 20px; text-align: center;">4</td> <td style="border: 1px solid black; width: 20px; text-align: center;">5</td> <td style="border: 1px solid black; width: 20px; text-align: center;">6</td> <td style="border: 1px solid black; width: 20px; text-align: center;">7</td> <td style="border: 1px solid black; width: 20px; text-align: center;">8</td> <td style="border: 1px solid black; width: 20px; text-align: center;">9</td> </tr> </table> </div>	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9											
0	1	2	3	4	5	6	7	8	9											

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` option is used and if we are currently in the question part (not in the answer sheet), or `<showcommand>{<arg1>}{<arg2>}` when all the boxes are to be produced.

```
1240 \newcommand\AMCformatChoices[4]{%
1241   \global\AMCrep@count=\z@%
1242   \AMC@if@separate@question{%
1243     \AMC@mem@add{\global\AMCrep@count=\z@%
1244       #1{#3}{#4}}%
1245   }%
1246   \ifAMC@ensemble%
1247     #2{#3}{#4}%
1248   \AMC@amclog{AUTOQCM[QPART]^J}%
1249   \else%
1250     #1{#3}{#4}%
1251   \fi%
1252 }
```

Some computation commands are now defined. The command `\amc_fp_decompose:NNn{<fp var>}{<int var>}{<x>}` sets `<fp var>` to be the *mantissa* and `<int var>` the *exponent* of the floating point number `<x>`. For example, `\amc_fp_decompose:NNn\mant_fp\expo_int{123.456}` give the value 1.23456 to `\mant_fp` and 2 to `\expo_int` (because  $123.456 = 1.23456 \times 10^2$ ).

The command `\amc_fp_to_digits:Nnnn{<clist>}{<x>}{<n digits>}{<base>}` rounds the floating point number `<x>` and populates the comma separated list `<clist>` with its `<n digits>` digits in base `<base>`. An error is issued if `<x>` would have required more digits.

```

1253 \ExplSyntaxOn
1254
1255 \cs_generate_variant:Nn \tl_replace_once:Nnn { Nxn }
1256
1257 \tl_new:N \amc_ee_tl
1258 \seq_new:N \amc_ee_seq

  Note that with some versions of l3fp-convert (prior to 2017-09-18), \fp_to_scientific leads to
  a ‘e’ with catcode 12 (other). We convert it to catcode letter before splitting.

1259 \group_begin:
1260 \char_set_catcode_other:N E
1261 \tl_to_lowercase:n
1262 {
1263   \cs_new:Npn \amc_read_scientific:NNn #1 #2 #3 {
1264     \tl_set:Nn \amc_ee_tl { #3 }
1265     \tl_replace_once:Nxn \amc_ee_tl { E } { e }
1266     \seq_set_split:NnV \amc_ee_seq e \amc_ee_tl
1267     \fp_set:Nn #1 { \seq_item:Nn \amc_ee_seq 1 }
1268     \int_set:Nn #2 { \seq_item:Nn \amc_ee_seq 2 }
1269   }
1270 }
1271 \group_end:
1272
1273 \cs_generate_variant:Nn \amc_read_scientific:NNn { NNf, NNx }
1274
1275 \fp_new:N \amc_fulls_fp
1276 \cs_new:Npn \amc_fp_decompose:NNn #1 #2 #3 {
1277   \fp_set:Nn \amc_fulls_fp { #3 }

  Note that with some versions of l3fp-convert, the exponent part is omitted for some values, so
  that we add e 0.

1278   \amc_read_scientific:NNx #1 #2
1279   { \fp_to_scientific:N \amc_fulls_fp e 0 }
1280 }
1281 \cs_generate_variant:Nn \amc_fp_decompose:NNn { NNx }
1282
1283 \fp_new:N \amc_num_mantissa_fp
1284 \int_new:N \amc_num_exponent_int
1285 \cs_new:Npn \amc_fp_n_significant_digits:Nnn #1 #2 #3 {
1286   \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_exponent_int
1287   { #2 }
1288   \fp_set:Nn #1
1289   { round(\amc_num_mantissa_fp * 10^((#3)-1)) }
1290   \fp_compare:nTF { abs(#1) >= 10^(#3) }
1291   {
1292     \fp_set:Nn #1 { #1 / 10 }
1293   } { }
1294 }
1295
1296 \fp_new:N \amc_num_sig_fp
1297 \cs_new:Npn \amc_fp_show_n_significant_digits:nn #1 #2 {

```

```

1298 \amc_fp_n_significant_digits:Nnn \amc_num_nsig_fp { #1 } { #2 }
1299 }
1300 \cs_new_eq:NN \AMCsignificantDigits \amc_fp_show_n_significant_digits:nn
1301
1302 \cs_new:Npn \amc_fp_show_significant_digits: {
1303   \fp_use:N \amc_num_nsig_fp
1304 }
1305 \cs_new_eq:NN \AMCshowSignificantDigits \amc_fp_show_significant_digits:
1306
1307 \cs_new:Npn \amc_fp_n_digits:Nnn #1 #2 #3 {
1308   \fp_set:Nn #1
1309     { round((#2) * 10^(#3)) }
1310 }
1311
1312 \int_new:N \amc_todigits_int
1313 \cs_new:Npn \amc_fp_to_digits:Nnnn #1 #2 #3 #4 {
1314   \clist_clear:N #1
1315   \int_set:Nn \amc_todigits_int { \fp_eval:n { abs(round(#2)) } }
1316   \prg_replicate:nn { #3 } {
1317     \clist_put_left:Nx #1 { \int_mod:nn \amc_todigits_int { #4 } }
1318     \int_set:Nn \amc_todigits_int
1319       { \int_div_truncate:nn \amc_todigits_int { #4 } }
1320   }
1321   \int_compare:nNnTF \amc_todigits_int = 0 { } {
1322     \message{^^J!~Error:~number~too~large,
1323       ~some~digits~will~be~discarded^^J}
1324   }
1325 }
1326
1327 \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:

```

1328 \def\AMCncontextGoto{}
1329 \def\AMCncontextVHead#1{\emph{b#1}}
1330 \newdimen\AMCnumeric@Hspace\AMCnumeric@Hspace=.5em
1331 \newdimen\AMCnumeric@Vspace\AMCnumeric@Vspace=1ex
1332 \ExplSyntaxOn
1333
1334 \keys_define:nn { amcnumeric } {
1335   Tsign .code:n = {\def\AMCncontextSign{#1}},
1336   Tsign .initial:n = {},
1337   Tpoint .code:n = {\def\AMCdecimalPoint{#1}},
1338   Tpoint .initial:n = { \raisebox{1ex}{\bf .} },
1339   Texponent .code:n = {\def\AMCexponent{#1}},
1340   Texponent .initial:n = { $\times 10$\textasciicircum },
1341   vspace .code:n = {\AMCnumeric@Vspace=#1},
1342   hspace .code:n = {\AMCnumeric@Hspace=#1},

```

```

1343 bordercol .code:n = {\def\AMCncol@Border{#1}},
1344 bordercol .initial:n = { lightgray },
1345 borderwidth .code:n = {\def\AMCncol@BorderWidth{#1}},
1346 borderwidth .initial:n = { 1mm },
1347 backgroundcol .code:n = {\def\AMCncol@Background{#1}},
1348 backgroundcol .initial:n = { white },
1349 digits .int_set:N = \amc_num_ndigits_int,
1350 digits .initial:n = { 3 },
1351 decimals .int_set:N = \amc_num_decd_int,
1352 decimals .initial:n = { 0 },
1353 exponent .int_set:N = \amc_num_expo_int,
1354 exponent .initial:n = { 0 },
1355 base .int_set:N = \amc_num_base_int,
1356 base .initial:n = { 10 },
1357 sign .bool_set:N = \amc_num_sign_bool,
1358 sign .initial:n = { true },
1359 sign .default:n = { true },
1360 exposign .bool_set:N = \amc_num_exposign_bool,
1361 exposign .initial:n = { true },
1362 exposign .default:n = { true },
1363 strict .bool_set:N = \amc_num_strict_bool,
1364 strict .initial:n = { false },
1365 strict .default:n = { true },
1366 scoring .bool_set:N = \amc_num_scoring_bool,
1367 scoring .initial:n = { true },
1368 scoring .default:n = { true },
1369 vertical .bool_set:N = \amc_num_vertical_bool,
1370 vertical .initial:n = { false },
1371 vertical .default:n = { true },
1372 expovertical .bool_set:N = \amc_num_expovertical_bool,
1373 expovertical .initial:n = { false },
1374 expovertical .default:n = { true },
1375 reverse .bool_set:N = \amc_num_reverse_bool,
1376 reverse .initial:n = { false },
1377 reverse .default:n = { true },
1378 vhead .bool_set:N = \amc_num_vhead_bool,
1379 vhead .initial:n = { false },
1380 vhead .default:n = { true },
1381 nozero .bool_set:N = \amc_num_nozero_bool,
1382 nozero .initial:n = { false },
1383 nozero .default:n = { true },
1384 significant .bool_set:N = \amc_num_significant_bool,
1385 significant .initial:n = { false },
1386 significant .default:n = { true },
1387 scoreexact .code:n = {\def\AMC@numeric@scoreexact{#1}},
1388 scoreexact .initial:n = { 2 },
1389 scoreapprox .code:n = {\def\AMC@numeric@scoreapprox{#1}},
1390 scoreapprox .initial:n = { 1 },
1391 exact .int_set:N = \amc_num_exact_int,
1392 exact .initial:n = { 0 },

```

```

1393 approx .int_set:N = \amc_num_approx_int,
1394 approx .initial:n = { 0 }
1395 }
1396
1397 \cs_new:Npn \amc_num_setopt #1 {
1398   \keys_set:nn { amcnumeric } { #1 }
1399 }
1400
1401 \cs_new_eq:NN \AMCnumericOpts \amc_num_setopt
1402

```

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

```

1403 \cs_new:Npn \amc_num_char:nn #1 #2 {
1404   \global\advance\AMCrep@count\one\relax
1405   \AMC@amclog{AUTOQCM[REP= \the\AMCrep@count :
1406     \ifx#2\AMC@checkbox B\else M\fi ]^^J}
1407   \ifAMC@correc
1408     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :
1409       \the\AMCid@quest , \the\AMCrep@count}
1410   \else
1411     \protect\AMC@keyBox@{#1}{-}{1}{case : \AMCid@name :
1412       \the\AMCid@quest , \the\AMCrep@count}
1413   \fi
1414 }

```

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.

```

1415 \cs_new:Npn \amc_num_digit_box:nn #1 #2 {
1416   \int_compare:nNnTF { #1 } = { #2 } {
1417     \amc_num_char:nn{ #1 }{\AMC@checkbox}
1418   } {
1419     \amc_num_char:nn{ #1 }{-}
1420   }
1421 }

```

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

```

1422 \cs_new:Npn \amc_num_sign_boxes:N #1 #2 {
1423   \bool_if:nTF { #1 } {
1424     \hbox{\amc_num_char:nn{+}{}}
1425     \vspace{\AMCnumeric@Vspace}
1426     \AMC@amclog{AUTOQCM[B=set. sign #2 =1]^^J}
1427     \hbox{\amc_num_char:nn{-}{\AMC@checkbox}}
1428     \AMC@amclog{AUTOQCM[B=set. sign #2 =-1]^^J}
1429   } {
1430     \hbox{\amc_num_char:nn{+}{\AMC@checkbox}}
1431     \vspace{\AMCnumeric@Vspace}
1432     \AMC@amclog{AUTOQCM[B=set. sign #2 =1]^^J}
1433     \hbox{\amc_num_char:nn{-}{}}

```



```

1434 \AMC@amclog{AUTOQCM[B=set. sign #2 =-1]^^J}
1435 }
1436 }

```

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 vaiable `<varname>` will be set to the value chosen by the student.

```

1437 \cs_new:Npn \amc_num_digit_boxes_h:nnn #1 #2 #3 {
1438   \int_step_inline:nnnn
1439   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1440   { 1 } { #3 - 1 } {
1441     \amc_num_digit_box:nn { ##1 }{ #2 }
1442     \AMC@amclog{AUTOQCM[B= set. #1 = ##1 ]^^J}
1443     \hspace{\AMCnumeric@Hspace}
1444   }
1445   \hspace{-\AMCnumeric@Hspace}
1446 }
1447
1448 \cs_new:Npn \amc_num_digit_boxes_v:nnn #1 #2 #3 {
1449   \int_step_inline:nnnn
1450   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1451   { 1 } { #3 - 1 } {
1452     \vbox{\hbox{
1453       \amc_num_digit_box:nn { ##1 }{ #2 }
1454     }}
1455     \AMC@amclog{AUTOQCM[B= set. #1 = ##1 ]^^J}
1456     \int_compare:nNnTF { ##1 } < { #3 - 1 } {
1457       \vspace{\AMCnumeric@Vspace}
1458     } {}
1459   }
1460 }
1461
1462 \int_new:N \amc_num_first_digit_int
1463 \cs_new:Npn \amc_num_digit_boxes_vr:nnn #1 #2 #3 {
1464   \int_set:Nn \amc_num_first_digit_int
1465   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1466   \int_step_inline:nnnn { #3 - 1 } { -1 }
1467   \amc_num_first_digit_int {
1468     \vbox{\hbox{
1469       \amc_num_digit_box:nn { ##1 }{ #2 }
1470     }}
1471     \AMC@amclog{AUTOQCM[B= set. #1 = ##1 ]^^J}
1472     \int_compare:nNnTF { ##1 } > \amc_num_first_digit_int {
1473       \vspace{\AMCnumeric@Vspace}
1474     } {}
1475   }
1476 }

```

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

```

1477 \cs_new:Npn \amc_num_integer_boxes_v:Nnn #1 #2 #3 {
    begin a loop over all digits,
1478   \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1479   \clist_map_inline:Nn #1 {
        place the decimal point if necessary,
1480       \int_compare:nNnTF \amc_num_digit_int = { #3 } {
1481           \hbox{ \AMCdecimalPoint } \hspace{\AMCnumeric@Hspace}
1482       } { }
        draw the box for this digit,
1483       \hbox{\vbox{
1484           \bool_if:NTF \amc_num_vhead_bool {
1485               \vbox{\hbox{\AMCtextVHead{ \int_eval:n
1486                   { \amc_num_digit_int - 1 } }}}
1487               \vspace{\AMCnumeric@Vspace}
1488           } { }
1489           \bool_if:NTF \amc_num_reverse_bool {
1490               \amc_num_digit_boxes_vr:nnn { #2
1491                   \int_to_Alph:n \amc_num_digit_int }
1492               { ##1 } { \amc_num_base_int }
1493           } {
1494               \amc_num_digit_boxes_v:nnn { #2
1495                   \int_to_Alph:n \amc_num_digit_int }
1496               { ##1 } { \amc_num_base_int }
1497           }
1498       }}
        and end the loop over digits, adding space if this is not the last one.
1499       \int_compare:nNnTF \amc_num_digit_int > 1 {
1500           \hspace{\AMCnumeric@Hspace}
1501       } { }
1502       \int_decr:N \amc_num_digit_int
1503   }
1504 }
1505

```

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

```

1506
1507 \cs_new:Npn \amc_num_integer_boxes_h:Nnn #1 #2 #3 {
1508   \vbox{
1509       \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1510       \clist_map_inline:Nn #1 {
1511           \int_compare:nNnTF
1512           \amc_num_digit_int = { #3 } {
1513               \hbox{ \AMCdecimalPoint }
1514           } { }
1515       \hbox{
1516           \amc_num_digit_boxes_h:nnn { #2
1517               \int_to_Alph:n \amc_num_digit_int }

```

```

1518      { ##1 } \amc_num_base_int
1519    }
1520    \int_compare:nNnTF \amc_num_digit_int > 1 {
1521      \vspace{\AMCnumeric@Vspace}
1522    } { }
1523    \int_decr:N \amc_num_digit_int
1524  }}
1525 }
1526

```

Finally, `\amc_num_integer_boxes:NnnNn{<correct digits>}{<prefix>}{<decimals>}{<sign bool>}{<positive>}` draws boxes for integer entry, including the sign if `<sign bool>` is true.

```

1527
1528 \cs_new:Npn \amc_num_integer_boxes:NnnNn #1 #2 #3 #4 #5 {
1529   \hbox{
1530     \bool_if:NTF { #4 } {
1531       \vbox{
1532         \ifx\AMCtextSign\@empty\@empty\else
1533           \hbox{\AMCtextSign}\vspace{\AMCnumeric@Vspace}\fi
1534           \amc_num_sign_boxes:N { #5 } { #2 }
1535         }
1536         \hspace{.5em}
1537         \vrule
1538         \hspace{.5em}
1539       } { }
1540     \hbox{
1541       \bool_if:NTF \amc_num_vertical_bool
1542       \amc_num_integer_boxes_v:Nnn \amc_num_integer_boxes_h:Nnn
1543       #1 { #2 } { #3 }
1544     }
1545   }
1546 }
1547

```

The command `\amc_num_build_integer_scoring:Nnnn{<tl var>}{<sign bool>}{<prefix>}{<n>}` builds a scoring to compute an integer from a serie of `<n>`-digits boxes, with name prefix `<prefix>`, using a sign variable if `<sign bool>` is true.

```

1548
1549 \cs_new:Npn \amc_num_build_integer_scoring:Nnnn #1 #2 #3 #4 {
1550   \tl_clear:N #1
1551   \int_set_eq:NN \amc_num_digit_int { #4 }
1552   \int_while_do:nNnn \amc_num_digit_int > 0 {
1553     \bool_if:NTF \amc_num_strict_bool {
1554       \AMC@amclog{AUTOQCM[B=requires. #3
1555         \int_to_Alph:n \amc_num_digit_int = 1]^~J}
1556     } {
1557       \AMC@amclog{AUTOQCM[B=default. #3
1558         \int_to_Alph:n \amc_num_digit_int = 0]^~J}
1559     }
1560   \int_compare:nNnTF \amc_num_digit_int = #4 { } {
1561     \tl_put_left:Nn #1 { ( }

```

```

1562     \tl_put_right:Nx #1 { ) *
1563     \int_use:N \amc_num_base_int + }
1564   }
1565   \tl_put_right:Nx #1
1566   { #3 \int_to_Alph:n \amc_num_digit_int }
1567   \int_decr:N \amc_num_digit_int
1568 }
1569 \tl_put_left:Nn #1 { ( }
1570 \tl_put_right:Nn #1 { ) }
1571 \bool_if:NTF { #2 } {
1572   \bool_if:NTF \amc_num_strict_bool {
1573     \AMC@amclog{AUTOQCM[B=requires. sign #3 =1]^^J}
1574   } {
1575     \AMC@amclog{AUTOQCM[B=default. sign #3 =1]^^J}
1576   }
1577   \tl_put_right:Nx #1 { * ( sign #3 ) }
1578 } { }
1579 }
1580

```

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

```

1581
1582 \fp_new:N \amc_num_correct_fp
1583 \clist_new:N \amc_num_digits_clist
1584 \clist_new:N \amc_num_expo_digits_clist
1585 \int_new:N \amc_num_digit_int
1586 \tl_new:N \amc_num_compute_tl
1587 \tl_new:N \amc_num_expo_tl
1588 \int_new:N \amc_num_correct_expo_int
1589
1590 \cs_new:Npn \amc_numeric_show:nn #1 #2 {

```

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

```

1591 \ifAMC@ensemble\ifAMCformulaire@dedans
1592   \AMC@amclog{AUTOQCM[Q=\the\AMCid@quest]^^J}
1593 \fi\fi

```

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

```

1594 { \keys_set:nn { amcnumeric } { #2 }
1595   \bool_if:nTF { \bool_if_p:N\amc_num_significant_bool
1596     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1597     \message{^^J!~AMCnumeric~Error:~significant=true~can't~be~used~with~base!=10.^^J}
1598   } {}
1599   \bool_if:nTF { \int_compare_p:n { \amc_num_expo_int != 0 }
1600     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1601     \message{^^J!~AMCnumeric~Error:~scientific~notation~can't~be~used~with~base!=10.^^J}
1602   } {}

```

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

```

1603   \bool_if:NTF \amc_num_significant_bool {
1604     \amc_fp_n_significant_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_ndigits_int

```

```

1605 } {
1606   \int_compare:nNnTF \amc_num_expo_int > 0 {
1607     \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_correct_expo_int { #1 }
1608     \int_compare:nNnTF { \amc_num_ndigits_int - \amc_num_decd_int } > 1 {
1609       \fp_set:Nn \amc_num_mantissa_fp {
1610         \amc_num_mantissa_fp * 10^( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1611       }
1612       \int_set:Nn \amc_num_correct_expo_int {
1613         \amc_num_correct_expo_int - ( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1614       }
1615     } {}
1616     \amc_fp_n_digits:Nnn \amc_num_correct_fp \amc_num_mantissa_fp \amc_num_decd_int
1617   } {
1618     \amc_fp_n_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_decd_int
1619   }
1620 }

```

Now extracts the required digits:

```

1621 \amc_fp_to_digits:Nnnn \amc_num_digits_clist \amc_num_correct_fp
1622   \amc_num_ndigits_int \amc_num_base_int
1623 \int_compare:nNnTF \amc_num_expo_int > 0 {
1624   \amc_fp_to_digits:Nnnn \amc_num_expo_digits_clist \amc_num_correct_expo_int
1625   \amc_num_expo_int \amc_num_base_int
1626 } {}

```

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.

```

1627 \bool_if:NTF \amc_num_scoring_bool {
1628   \AMC@amclog{AUTOQCM[B=haut=mz=,
1629     formula=(Vdifference <= \int_use:N \amc_num_exact_int ?
1630     \AMC@numeric@scoreexact :
1631     \int_compare:nNnTF \amc_num_approx_int = 0 {
1632       0
1633     } {
1634       (Vdifference <= \int_use:N\amc_num_approx_int ?
1635       \AMC@numeric@scoreapprox : 0)
1636     }
1637   ]^^J}
1638 } {}
1639 \amc_num_build_integer_scoring:Nnnn
1640   \amc_num_compute_tl \amc_num_sign_bool { digit } \amc_num_ndigits_int
1641 \int_compare:nNnTF \amc_num_expo_int > 0 {
1642   \amc_num_build_integer_scoring:Nnnn
1643     \amc_num_expo_tl \amc_num_exposign_bool { expo } \amc_num_expo_int
1644     \AMC@amclog{AUTOQCM[B= set. intE = \amc_num_expo_tl ]^^J}
1645     \tl_put_right:Nx \amc_num_compute_tl
1646       { * \int_use:N\amc_num_base_int ** ( intE - ( \int_use:N\amc_num_correct_expo_int ) ) }
1647   } {}
1648 \AMC@amclog{AUTOQCM[B= set.intV = \fp_to_int:N\amc_num_correct_fp ,
1649   set.intX = \amc_num_compute_tl ]^^J}
1650 \bool_if:NTF \amc_num_significant_bool {

```

```

1651 \AMC@amclog{AUTOQCM[B=set.Vdifference="min( abs((intV)-(intX)) ,
1652     abs(\int_use:N\amc_num_base_int * (intV) - (intX)) ,
1653     abs((intV) - \int_use:N\amc_num_base_int * (intX)) )"]^^J}
1654 } {
1655 \AMC@amclog{AUTOQCM[B=set.Vdifference=abs((intV)-(intX))]}^^J}
1656 }

```

Begin now with the frame around all the boxes:

```

1657 \vspace{1.5ex}\par{
1658 \fboxrule=\AMCncol@BorderWidth
1659 \fcolorbox{\AMCncol@Border}{\AMCncol@Background}{
1660 \bool_if:NTF \amc_num_expovertical_bool {
1661 \hbox{\vbox{
1662 \vbox{\amc_num_integer_boxes:NnnNn
1663 \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1664 { \fp_compare_p:nNn \amc_num_correct_fp < 0}}
1665 \int_compare:nNnTF \amc_num_expo_int > 0 {
1666 \vspace{\AMCnumeric@Vspace}
1667 \vbox{\hbox{\AMCexponent}}
1668 \vspace{\AMCnumeric@Vspace}
1669 \vbox{\amc_num_integer_boxes:NnnNn
1670 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
1671 { \int_compare_p:nNn \amc_num_correct_expo_int < 0 }}
1672 } {}
1673 }}
1674 } {
1675 \amc_num_integer_boxes:NnnNn
1676 \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1677 { \fp_compare_p:nNn \amc_num_correct_fp < 0}
1678 \int_compare:nNnTF \amc_num_expo_int > 0 {
1679 \hspace{\AMCnumeric@Hspace}\AMCexponent\hspace{\AMCnumeric@Hspace}
1680 \amc_num_integer_boxes:NnnNn
1681 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
1682 { \int_compare_p:nNn \amc_num_correct_expo_int < 0 }
1683 } {}
1684 }
1685 }
1686 }

```

And tell AMC that we finished with this question:

```

1687 \ifAMC@ensemble\else\vspace{1.5ex}\par\fi
1688 \ifAMC@ensemble\ifAMCformulaire@dedans
1689 \AMC@amclog{AUTOQCM[FQ]}^^J}
1690 \fi\fi
1691 }
1692 }
1693
1694 \cs_new_eq:NN \AMCnumericShow \amc_numeric_show:nn
1695

```

`\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.

```

1696 \cs_new:Npn \amc_numeric_hide:nn #1 #2 {
1697   \keys_set:nn { amcnumeric } { #2 }
1698   \AMCncontextGoto
1699   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi
1700 }
1701
1702 \cs_new_eq:NN \AMCnumericHide \amc_numeric_hide:nn
1703
1704 \ExplSyntaxOff
1705 \def\AMCnumericChoicesPlain{%
1706   \AMC@if@separate@question{\AMC@mem@category{numeric}}%
1707   \AMCformatChoices{\AMCnumericShow}{\AMCnumericHide}%
1708 }

```

The  $\langle value \rangle$  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...

```

1709 \ExplSyntaxOn
1710
1711 \cs_new:Npn \amc_numeric_choices:nn #1#2 {
1712   \AMCnumericChoicesPlain{#1}{#2}
1713 }
1714 \cs_generate_variant:Nn \amc_numeric_choices:nn { xn }
1715 \cs_new_eq:NN \AMCnumericChoices \amc_numeric_choices:xn
1716
1717 \ExplSyntaxOff

```

### 4.13.3 Intervals

`\AMCIntervals` The command `\AMCIntervals{ $\langle x \rangle$ }{ $\langle x0 \rangle$ }{ $\langle x1 \rangle$ }{ $\langle \delta \rangle$ }` 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 4** 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

```

1718 \def\AMCIntervalFormat#1#2{[#1,\,#2]}
to follow local conventions (writting  $[a, b)$  instead of  $[a, b[$  is for example a common usage).
1719 \ExplSyntaxOn
1720
1721 \fp_new:N \amc_interv_a
1722 \fp_new:N \amc_interv_b
1723 \cs_new:Npn \amc_intervals:nnnn #1 #2 #3 #4 {
1724   \fp_set:Nn \amc_interv_a { #2 }
1725   \fp_do_while:nn { \amc_interv_a < #3 } {
1726     \fp_set:Nn \amc_interv_b { \amc_interv_a + #4 }
1727     \fp_compare:nTF { \amc_interv_a <= #1 < \amc_interv_b }
1728       \correctchoice \wrongchoice
1729     {\AMCIntervalFormat{\fp_use:N \amc_interv_a}{\fp_use:N \amc_interv_b}}
1730     \fp_set:Nn \amc_interv_a \amc_interv_b
1731   }
1732 }
1733 \cs_new_eq:NN \AMCIntervals \amc_intervals:nnnn
1734
1735 \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 5**    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:

```

1736 \def\AMCotextGoto{}
1737 \def\AMCotextReserved{}
1738 \def\AMCocol@Background{lightgray}
1739 \def\AMCocol@BoxFrameRule{white}
1740 \def\AMCocol@FrameRule{black}
1741 \def\AMCocol@Foreground{}
1742 \def\AMCopen@answer{}
1743 \def\AMCopen@question{}
1744 \def\AMCopen@lineuptext{}
1745 \define@key{AMCopen}{backgroundcol}{\def\AMCocol@Background{#1}}
1746 \define@key{AMCopen}{foregroundcol}{\def\AMCocol@Foreground{#1}}
1747 \define@key{AMCopen}{Treserved}{\def\AMCotextReserved{#1}}
1748 \define@key{AMCopen}{question}{\AMCid@name}{\def\AMCopen@question{#1}}
1749 \define@key{AMCopen}{answer}{\def\AMCopen@answer{#1}}
1750 \define@key{AMCopen}{contentcommand}{\AMCopen@lines}{\def\AMCopen@contentcommand{#1}}
1751 \newdimen\AMCopen@Hspace\AMCopen@Hspace=.5em
1752 \define@key{AMCopen}{hspace}{\AMCopen@Hspace=#1}
1753 \def\AMCopen@Width{.95\linewidth}
1754 \define@key{AMCopen}{width}{\def\AMCopen@Width{#1}}
1755 \newdimen\AMCopen@LineHeight\AMCopen@LineHeight=1cm
1756 \define@key{AMCopen}{lineheight}{\AMCopen@LineHeight=#1}
1757 \newcount\AMCopen@Lines\AMCopen@Lines=1
1758 \define@key{AMCopen}{lines}{\AMCopen@Lines=#1}
1759 \newdimen\AMCopen@boxmargin\AMCopen@boxmargin=3pt
1760 \define@key{AMCopen}{boxmargin}{\AMCopen@boxmargin=#1}
1761 \newdimen\AMCopen@boxframerule\AMCopen@boxframerule=1pt
1762 \define@key{AMCopen}{boxframerule}{\AMCopen@boxframerule=#1}
1763 \define@key{AMCopen}{boxframerulecol}{\def\AMCocol@BoxFrameRule{#1}}
1764 \define@key{AMCopen}{framerulecol}{\def\AMCocol@FrameRule{#1}}
1765 \newdimen\AMCopen@framerule\AMCopen@framerule=1pt
1766 \define@key{AMCopen}{framerule}{\AMCopen@framerule=#1}
1767 \define@key{AMCopen}{lineuptext}{\def\AMCopen@lineuptext{#1}}
1768 \define@boolkey{AMCopen}{dots}[true]{}
1769 \define@boolkey{AMCopen}{scan}[true]{}
1770 \define@boolkey{AMCopen}{annotate}[false]{}
1771 \define@boolkey{AMCopen}{lineup}[false]{}
1772 \setkeys{AMCopen}{dots,scan,annotate,lineup,contentcommand}
1773 \newcommand\AMCopenOpts[1]{\setkeys{AMCopen}{#1}}

```

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

```

1774 \newcommand\AMCopen@lines{%
1775   \begin{minipage}{\AMCopen@Width}%
1776     \loop\vspace{\AMCopen@LineHeight}
1777     \hspace*{.5em}\ifAMC@correc\smash{\AMCopen@answer}\def\AMCopen@answer{}\fi%
1778     \ifKV@AMCopen@dots%
1779     \dotfill\hspace*{.5em}
1780   \fi

```

```

1781 \ifnum\AMCopen@Lines>\@ne\par\advance\AMCopen@Lines\m@ne\repeat%
1782 \end{minipage}
1783 }
1784 \newcommand\AMCopenShow[2]{
1785 \ifAMC@ensemble\ifAMCformulaire@dedans%
1786 \AMC@amclog{AUTOQCM[Q=\the\AMCid@quest]^^J}%
1787 \fi\fi%
1788 {\setkeys{AMCopen}{#1}%
1789 \ifKV@AMCopen@lineup%
1790 \ifAMC@ensemble\else%
1791 \ifx\@empty\AMCopen@lineuptext\@empty\fi%
1792 \fi%
1793 \ifAMC@correc\smash{\AMCopen@answer}\fi%
1794 \ifx\@empty\AMCopen@lineuptext\@empty%
1795 \dotfill%
1796 \else%
1797 \AMCopen@lineuptext\hfill%
1798 \fi%
1799 \else%
1800 \hspace*{.5em}\linebreak[1]\hspace*{\fill}%
1801 \fi%
1802 {\AMCnoCompleteMulti%
1803 \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
1804 \def\AMCanswer##1##2{\ifAMC@ensemble ##1\else%
1805 \ifAMC@inside@box ##1\else{\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
1806 \hspace{\AMCopen@Hspace}}%
1807 \fbboxsep=\AMCopen@boxmargin%
1808 \fbboxrule=\AMCopen@boxframerule%
1809 \fcolorbox{\AMCocol@BoxFrameRule}{\AMCocol@Background}{%
1810 \ifAMC@ensemble\AMCopen@question%
1811 \ifx\@empty\AMCopen@question\@empty\else\hspace{\AMCopen@Hspace}\fi%
1812 \fi%
1813 \begin{choicescustom}[o]%
1814 \ifx\AMCocol@Foreground\@empty\@empty\else%
1815 \def\AMC@boxcolor{\AMCocol@Foreground}%
1816 \fi%
1817 #2%
1818 \ifKV@AMCopen@scan\else\AMCdontScan\fi%
1819 \ifKV@AMCopen@annotate\else\AMCdontAnnotate\fi%
1820 \end{choicescustom}%
1821 \ifx\@empty\AMCotextReserved\@empty%
1822 \hspace{-\AMCopen@Hspace}%
1823 \else%
1824 \ifx\AMCocol@Foreground\@empty\@empty%
1825 \AMCotextReserved%
1826 \else%
1827 \textcolor{\AMCocol@Foreground}{\AMCotextReserved}%
1828 \fi%
1829 \fi%
1830 }%

```

```

1831 \ifKV@AMCOpen@lineup\else%
1832 \par\nobreak\noindent%
1833 \hspace*{\fill}{%
1834 \fboxrule=\AMCopen@framerule%
1835 \fcolorbox{\AMCocol@FrameRule}{white}{%
1836 \csname\AMCopen@contentcommand\endcsname
1837 }}%
1838 \vspace{\AMCpostOquest}\par%
1839 \fi%
1840 }%
1841 \ifAMC@ensemble\ifAMCformulaire@dedans%
1842 \AMC@amclog{AUTOQCM[FQ]^^J}%
1843 \fi\fi%
1844 }
1845 \newcommand\AMCopenHide[2]{%
1846 \AMCotextGoto%
1847 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
1848 }
1849 \def\AMCOpen{%
1850 \AMC@if@separate@question{\AMC@mem@category{open}}%
1851 \AMCformatChoices{\AMCopenShow}{\AMCopenHide}%
1852 }

```

#### 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}

1853 \def\AMCbotextGoto{}
1854 \def\AMCbo@help{}
1855 \define@key{AMCBoxOnly}{help}{\def\AMCbo@help{#1}}
1856 \define@boolkey{AMCBoxOnly}{ordered}[false]{}
1857 \setkeys{AMCBoxOnly}{ordered}
1858 \newcommand\AMCboOpts[1]{\setkeys{AMCBoxOnly}{#1}}
1859 \newcommand\AMCboShow[2]{%
1860 \ifAMC@ensemble\ifAMCformulaire@dedans%
1861 \AMC@amclog{AUTOQCM[Q=\the\AMCid@quest]^^J}%
1862 \fi\fi%
1863 {\setkeys{AMCBoxOnly}{#1}%
1864 \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
1865 \def\AMCanswer##1##2{\hspace{\AMCformHSpace} \ifAMC@ensemble ##1\else%
1866 \ifAMC@inside@box ##1\else\AMCboxOutsideLetter{##1}{##2}\fi\fi%
1867 }%

```

```

1868 \ifAMC@ensemble\AMCbo@help\fi%
1869 \ifKV@AMCBoxOnly@ordered%
1870 \begin{choicescustom}[o]%
1871 \else%
1872 \begin{choicescustom}%
1873 \fi%
1874 #2
1875 \end{choicescustom}%
1876 }%
1877 \ifAMC@ensemble\ifAMCformulaire@dedans%
1878 \AMC@amclog{AUTOQCM[FQ]^^J}%
1879 \fi\fi%
1880 }
1881 \newcommand\AMCboHide[2]{
1882 \AMCbotextGoto%
1883 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
1884 }
1885 \def\AMCBoxOnly{%
1886 \AMC@if@separate@question{\AMC@mem@category{box}}}%
1887 \AMCformatChoices{\AMCboShow}{\AMCboHide}%
1888 }

```

## 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 printing old versions of the subject.

```

1889 \DeclareFontShape{OT1}{cmr}{b}{n}{<35->cmr17}{-}
1890 \def\AMC@watertext{\AMC@loc@draft}
1891 \newcommand\AMCw@termark{%
1892 \setlength{\@tempdimb}{.5\paperwidth}%
1893 \setlength{\@tempdimc}{-.5\paperheight}%
1894 \put(\strip@pt\@tempdimb,\strip@pt\@tempdimc){%
1895 \makebox(0,0){\rotatebox{45}{\AMC@LR{%
1896 \textcolor[gray]{0.8}{
1897 \fontencoding{OT1}\fontfamily{cmr}
1898 \fontseries{b}\fontshape{n}
1899 \fontsize{90pt}{120pt}
1900 \selectfont
1901 \AMC@watertext}}}}}
1902 \newcommand\AMCw@terprint[1]{%
1903 \setbox\@tempboxa\vbox to \z@{%
1904 \vbox{%
1905 \hbox to \z@{%
1906 #1\hss}}\vss}
1907 \dp\@tempboxa\z@
1908 \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}.

```

1909 \def\AMCcircle#1#2{%
1910   {\setlength{\unitlength}{1mm}%
1911     \begin{picture}(\unitlength,\unitlength)(-#2,-#2)\thinlines\circle*{#1}\end{picture}}
1912 \def\m@rqueCalage{\AMCcircle{3.6}{1.8}}
1913 \def\m@rque#1{\AMC@tracebox{1}{#1}{\m@rqueCalage}}
1914 \def\he@dtaille#1{\vbox to 1cm{#1}}
1915 \def\he@dbas#1{\he@dtaille{\vspace*{\fill}#1}}
1916 \def\he@dhaut#1{\he@dtaille{#1\vspace*{\fill}}}
1917 \def\AMC@intituleHead{\AMC@loc@corrected}
1918 \def\AMC@note{}
1919 \def\AMCsetFoot#1{\def\AMC@note{#1}}
1920 \newcommand\AMCStudentNumber{\the\AMCid@etud}
1921 \newcommand\AMCIDBoxesA{\AMCbin@begin{1}\AMC@binaryBoxes[\AMC@NCBetud]{\the\AMCid@etud}}
1922 \newcommand\AMCIDBoxesB{\AMCbin@begin{2}\AMC@binaryBoxes[\AMC@NCBpage]{\thepage}}
1923 \newcommand\AMCIDBoxesC{\AMCbin@begin{3}\AMC@binaryBoxes[\AMC@NCBcheck]{\the\AMCid@check}}
1924 \newcommand\AMCIDBoxesABC{%
1925   \hbox{\vbox{\noindent\AMCIDBoxesA\
1926     \noindent\AMCIDBoxesB\AMCIDBoxesC}}}%
1927 }
1928 \AtBeginPage{\ifAMC@pagelayout\global\advance\AMCid@check\m@ne%
1929   \ifnum\AMCid@check<1\global\AMCid@check=\AMCid@checkmax\fi%
1930   \AMC@pagepos%
1931   \ifAMC@watermark\ifAMC@correthead\else\AMCw@terprint{\AMCw@termark}%
1932     \fi\fi\fi}
1933 \fancypagestyle{AMCpageHeadOnly}{%
1934   \fancyhf{}\fancyhead[C]{\textsc{\AMC@intituleHead}}%
1935   \renewcommand{\headrulewidth}{0pt}%
1936   \renewcommand{\footrulewidth}{0pt}%
1937 }
1938 \fancypagestyle{AMCpageFull}{%
1939   \fancyhf{}%
1940   \fancyhead[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHG}}}}%
1941   \fancyhead[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHD}}}}%
1942   \fancyfoot[L]{\AMC@LR{\leavevmode\m@rque{positionBG}}}%
1943   \fancyfoot[R]{\AMC@LR{\leavevmode\m@rque{positionBD}}}%
1944   \fancyhead[C]{\AMC@LR{\he@dhaut{%
1945     \begin{minipage}[b]{\AMC@CBtaille}\AMCboxColor{black}%
1946       \ifAMCids@top\vbox to \AMCids@height{\texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
1947       \AMCIDBoxesABC
1948     \end{minipage}}%
1949     \ifAMCids@side\hbox to \AMCids@width{\hspace*{\fill}}%

```

```

1950      \texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
1951    }}}}
1952    \fancyhfoffset[EOLR]{5mm}%
1953    \fancyfoot[C]{\AMC@note}%
1954    \renewcommand{\headrulewidth}{0pt}%
1955    \renewcommand{\footrulewidth}{0pt}%
1956  }
1957  \newcommand\AMCsubjectPageTag{%
1958    \fbox{\texttt{\the\AMCid@etud:\thepage}}%
1959  }
1960  \fancypagestyle{AMCpageNoMarks}{%
1961    \fancyhf{}%
1962    \fancyhead[R]{\AMCsubjectPageTag}%
1963    \fancyfoot[C]{\AMC@note}%
1964    \renewcommand{\headrulewidth}{0pt}%
1965    \renewcommand{\footrulewidth}{0pt}%
1966  }
1967  \fancypagestyle{AMCpageEmpty}{%
1968    \fancyhf{}%
1969    \renewcommand{\headrulewidth}{0pt}%
1970    \renewcommand{\footrulewidth}{0pt}%
1971  }
1972  \AtBeginDocument{%
1973    \ifAMC@pagelayout%
1974      \ifAMC@correthead
1975        \pagestyle{AMCpageHeadOnly}
1976      \else
1977        \pagestyle{AMCpageFull}
1978      \fi
1979    \fi
1980  }

```

#### 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.

```

1981 \newcommand{\onecopy}[2]{%
1982   \ifx\AMCNombreCopies\undefined\AMCnum@copies=#1%
1983   \else\AMCnum@copies=\AMCNombreCopies\fi%
1984   \AMC@amclog{AUTOQCM[TOTAL=\the\AMCnum@copies]^^J}%
1985   \AMCid@etud=\AMCid@etudstart%
1986   \ifnum\AMCid@etud=0\AMCid@etud=\AMC@premierecopie\fi%
1987   \AMCid@etudfin=\AMCnum@copies%
1988   \advance\AMCid@etudfin\AMCid@etud\relax%
1989   \ifAMC@correthead\AMCid@etudfin=\AMC@premierecopie\fi
1990   \ifAMC@pdfform\begin{Form}\fi%
1991   \loop{%
1992     \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
1993       \string\rngstate{\the\AMCid@etud}{\the\AMC@SR}%

```

```

1994     }\fi%
1995     \AMC@zoneformulairefalse\setcounter{page}{1}\setcounter{section}{0}%
1996     \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageNoMarks}\fi\fi%
1997     \AMCnumero{1}%
1998     \ifAMC@calibration\AMC@amclog{AUTOQCM[ETU=\the\AMCid@etud]^^J}\fi%
1999     \AMC@keepmemoryfalse%
2000     #2%
2001     \ifAMC@keepmemory\else\AMC@mem@clear\fi%
2002     \clearpage}%
2003 \advance\AMCid@etud\@ne\ifnum\AMCid@etud<\AMCid@etudfin\repeat%
2004 \ifAMC@pdfform\end{Form}\fi%
2005 \global\AMCid@etudstart=\AMCid@etud%
2006 }

```

**\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  $\langle n \rangle$  pages in the current question sheet.

```

2007 \newcount\AMC@addpages
2008 \newcommand{\AMCaddpagesto}[1]{%
2009   \AMC@addpages=#1\advance\AMC@addpages\@ne%
2010   \clearpage%
2011   \@whilenum\thepage<\AMC@addpages\do{%
2012     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2013     \hbox{}\clearpage%
2014   }%
2015 }

```

**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).

```

2016 \def\AMCcleardoublepage{%
2017   \clearpage%
2018   \ifodd\thepage\else%
2019     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2020     \hbox{}\clearpage%
2021   \fi%
2022 }

```

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

```

2023 \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` to be able to use different labels for different sheets.

```

\AMCref 2024 \newcommand\AMCstudentlabel[1]{\the\AMCid@etud-#1}
2025 \def\AMClabel#1{\expandafter\label{\AMCstudentlabel{#1}}}
2026 \def\AMCref#1{\expandafter\ref{\AMCstudentlabel{#1}}}
2027 \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.

```
2028 \newcommand{\AMCqlabel}[1]{%
2029   \AMClabel{#1}%
2030 }
```

## 4.18 Pre-association

`\AMCassociation` Association between sheets and students can be made before the exam with the `\AMCassociation{<id>}` command.

```
2031 \newcommand{\AMCassociation}[1]{%
2032   \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
2033     \string\association{the\AMCid@etud}{#1}%
2034   }\fi%
2035 }
```

## 4.19 Package options

See section 3.1 for the options descriptions.

```
2036 \def\AMC@lang@code{}
2037 \DeclareOptionX{noshuffle}{\AMC@ordretrue}
2038 \DeclareOptionX{noshufflegroups}{\AMC@shuffleGfalse}
2039 \DeclareOptionX{fullgroups}{\AMC@fullGroupstrue}
2040 \DeclareOptionX{answers}{\AMC@corretheadtrue\AMC@correcttrue}
2041 \DeclareOptionX{indivanswers}{\AMC@correcttrue}
2042 \DeclareOptionX{box}{\AMC@qbloctrue}
2043 \DeclareOptionX{asbox}{\AMC@asqbloctrue}
2044 \DeclareOptionX{separateanswersheet}{\AMC@ensembletrue}
2045 \DeclareOptionX{digits}{\AMC@inside@digittrue}
2046 \DeclareOptionX{ordre}{\AMC@ordretrue}
2047 \DeclareOptionX{correc}{\AMC@corretheadtrue\AMC@correcttrue}
2048 \DeclareOptionX{modele}{\AMC@corretheadtrue\AMC@correcfalse\AMC@ordretrue}
2049 \DeclareOptionX{correcindiv}{\AMC@correcttrue}
2050 \DeclareOptionX{init}{\AMC@SR@time}
2051 \DeclareOptionX{bloc}{\AMC@qbloctrue}
2052 \DeclareOptionX{completemulti}{\AMC@complete@multitrue}
2053 \DeclareOptionX{insidebox}{\AMC@inside@boxtrue}
2054 \DeclareOptionX{ensemble}{\AMC@ensembletrue}
2055 \DeclareOptionX{chiffres}{\AMC@inside@digittrue}
2056 \DeclareOptionX{outsidebox}{\AMC@outside@boxtrue}
2057 \DeclareOptionX{calibration}{\AMC@calibrationtrue}
2058 \DeclareOptionX{nowatermark}{\AMC@watermarkfalse}
2059 \newcommand\AMC@catalogMode{%
2060   \AMC@watermarkfalse\AMC@corretheadtrue%
2061   \AMC@correcttrue\AMC@ordretrue\AMC@shuffleGfalse%
2062   \AMC@fullGroupstrue%
2063   \def\AMC@intituleHead{\AMC@loc@catalog}\AMC@affichekeystrue}
2064 \DeclareOptionX{catalog}{\AMC@catalogMode}
```



```

2065 \DeclareOptionX{francais}{\def\AMC@lang@code{FR}\AMC@loc@FR}
2066 \DeclareOptionX{lang}{\def\AMC@lang@code{#1}\csname AMC@loc@#1\endcsname}
2067 \DeclareOptionX{versionA}{%
2068   \def\AMCid@checkmax{31}\def\AMC@NCBetud{9}\def\AMC@NCBpage{4}%
2069   \def\AMC@NCBcheck{5}\setlength{\AMC@CBtaille}{4cm}%
2070   \def\AMC@premierecopie{100}}
2071 \DeclareOptionX{plain}{\AMC@plaintrue}
2072 \DeclareOptionX{nopage}{\AMC@pagelayoutfalse}
2073 \DeclareOptionX{postcorrect}{\AMC@postcorrecttrue}
2074 \DeclareOptionX{automarks}{\AMC@automarkstrue}
2075 \newif\ifAMCneeds@storebox\AMCneeds@storeboxfalse
2076 \DeclareOptionX{storebox}{\AMCneeds@storeboxtrue}
2077 \DeclareOptionX{pdfform}{\AMC@pdfformtrue}
2078 \ProcessOptionsX
2079
2080 \ifAMCneeds@storebox
2081   \RequirePackage{storebox}\AtBeginDocument{{}}%
2082 \fi
2083 \ifAMC@pdfform
2084   \AMC@amclog{AUTOQCM[VAR:project:pdfform=1]^^J}%
2085   \AMC@boxStyle{shape=form}%
2086   \RequirePackage[pageanchor=false]{hyperref}%
2087 \else%
2088   \AMC@amclog{AUTOQCM[VAR:project:pdfform=0]^^J}%
2089 \fi
2090 \AtBeginDocument{
2091   \ifAMCneeds@storebox
2092     \let\AMC@new@savebox=\newstorebox%
2093     \let\AMC@save@box=\storebox%
2094     \let\AMC@use@box=\usestorebox%
2095   \fi
2096   \AMC@new@savebox{\AMC@ovalbox@R}
2097   \AMC@new@savebox{\AMC@ovalbox@RF}
2098   \AMC@new@savebox{\AMC@ovalbox@O}
2099   \AMC@new@savebox{\AMC@ovalbox@F}
2100 }

```

## 4.20 Package Errors

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

```

2101 \def\AMC@error@explain{\PackageError{automultiplechoice}{
2102   Command \protect\explain\space can only be used inside\MessageBreak question like environments}{Something
2103 }}

```

## 4.21 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.

```

2104 \ifAMC@plain

```

```

2105 \else
2106   \IfFileExists{environ.sty}{\RequirePackage{environ}}{}
2107   \ifx\TeXversion\@undefined
2108     \else
2109       \RequirePackage{etex}
2110     \fi
2111 \fi

```

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

Environment `{examcopy}[<n>]` does the same as command `onecopy`: it encloses  $\text{\LaTeX}$  code which makes *one* exam copy. Optional argument `<n>` gives the number of desired copies – this can also be modified redefining `\AMCNombreCopies`.

```

2112 \@ifpackageloaded{environ}{%
2113   \ifx\NewEnviron\undefined\PackageWarning{automultiplechoice}%
2114     {Package environ loaded but too old version:
2115      environnement examcopy/copieexamen will NOT be defined.}%
2116   \else\NewEnviron{examcopy}[1][5]{\onecopy{#1}{\BODY}}\fi}%
2117 {\PackageWarning{automultiplechoice}%
2118   {Package environ not loaded: environnement
2119    examcopy/copieexamen will NOT be defined.}}

```

## 4.22 External control

`\SujetExterne` 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  $\text{\LaTeX}$  file:

`\ScoringExterne` `\CorrigeExterne` `\CorrigeIndivExterne` `\NoWatermarkExterne` `pdflatex '\nonstopmode\def\SujetExterne{1}\def\NoWatermarkExterne{1}\input{mcq.tex}'`

```

2120 \ifx\SujetExterne\undefined\else
2121 \message{***SUJET***^^J}
2122 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse
2123 \fi
2124 \ifx\ScoringExterne\undefined\else
2125 \message{***SCORING***^^J}
2126 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse\AMC@invisibletrue
2127 \fi
2128 \ifx\CorrigeExterne\undefined\else
2129 \message{***CORRIGE***^^J}
2130 \AMC@calibrationfalse\AMC@corretheadtrue\AMC@correcttrue\AMC@watermarkfalse
2131 \fi
2132 \ifx\CorrigeIndivExterne\undefined\else
2133 \message{***CORRIGE***^^J}
2134 \AMC@calibrationfalse\AMC@corretheadfalse\AMC@correcttrue\AMC@watermarkfalse
2135 \fi
2136 \ifx\CatalogExterne\undefined\else
2137 \message{***CATALOG***^^J}
2138 \AMC@catalogMode
2139 \fi
2140 \ifx\NoWatermarkExterne\undefined\else

```

```

2141 \AMC@watermarkfalse
2142 \fi

```

## 4.23 Page layout

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

```

2143 \ifpackage{geometry}{\usepackage{geometry}}
2144 \ifAMC@pagelayout
2145   \ifAMC@correthead
2146     \geometry{hmargin=3cm,vmargin={1cm,1cm},includeheadfoot,headheight=1cm,footskip=1cm}
2147   \else
2148     \geometry{hmargin=3cm,headheight=2cm,headsep=.3cm,footskip=1cm,top=3.5cm,bottom=2.5cm}
2149   \fi
2150   \ifAMC@watermark
2151     \ifAMC@correthead\else
2152       \def\AMC@note{\begin{minipage}{0.65\linewidth}
2153         \AMC@LR{\textcolor{blue}{\AMC@loc@message}}
2154       \end{minipage}}
2155     }
2156   \fi
2157 \fi
2158 \fi

```

## 4.24 Initialisation

Initialisation of the check counter:

```

2159 \AMCid@check=\AMCid@checkmax\advance\AMCid@check\@ne

```

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

```

2160 \ifAMC@ensemble\AMC@amclog{AUTOQCM[VAR:ensemble=1]^^J}\fi
2161 \ifAMC@inside@box\AMC@amclog{AUTOQCM[VAR:insidebox=1]^^J}\fi
2162 \ifAMC@outside@box\AMC@amclog{AUTOQCM[VAR:outsidebox=1]^^J}\fi
2163 \ifAMC@postcorrect\AMC@amclog{AUTOQCM[VAR:postcorrect=1]^^J}\fi

```

Preparing writing to .xy file :

```

2164 \ifAMC@calibration
2165 \newwrite\AMC@XYFILE%
2166 \immediate\openout\AMC@XYFILE\jobname.xy%
2167 \immediate\write\AMC@XYFILE{\string\version{\AMC@VERSION}}
2168 \immediate\write\AMC@XYFILE{\string\with{codedigit=squarebrackets}}
2169 \immediate\write\AMC@XYFILE{\string\with{version=\AMC@VERSION}}
2170 \immediate\write\AMC@XYFILE{\string\with{ensemble=\ifAMC@ensemble yes\else no\fi}}
2171 \immediate\write\AMC@XYFILE{\string\with{insidebox=\ifAMC@inside@box yes\else no\fi}}
2172 \immediate\write\AMC@XYFILE{\string\with{outsidebox=\ifAMC@outside@box yes\else no\fi}}
2173 \immediate\write\AMC@XYFILE{\string\with{postcorrect=\ifAMC@postcorrect yes\else no\fi}}
2174 \immediate\write\AMC@XYFILE{\string\with{lang=\AMC@lang@code}}
2175 \ifx\AMCNombreCopies\undefined%
2176 \immediate\write\AMC@XYFILE{\string\with{ncopies=default}}}%
2177 \else%

```

```

2178 \immediate\write\AMC@XYFILE{\string\with{ncopies=\AMCNombreCopies}}}%
2179 \fi%
2180 \fi

```

## 4.25 French command names

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

```

2181 \let\reponses=\choices\let\endreponses=\endchoices
2182 \let\reponseshoriz=\choiceshoriz\let\endreponseshoriz=\endchoiceshoriz
2183 \let\reponsesperso=\choicescustom\let\endreponsesperso=\endchoicescustom
2184 \let\bonne=\correctchoice
2185 \let\mauvaise=\wrongchoice
2186 \let\bareme=\scoring
2187 \let\baremeDefaultM=\scoringDefaultM
2188 \let\baremeDefaultS=\scoringDefaultS
2189 \def\exemplaire{\AMC@loc@FR\onecopy}
2190 \ifpackageloaded{environ}{%
2191   \let\copieexamen=\examcopy\let\endcopieexamen=\endexamcopy}{%
2192   \let\melangegroupe=\shufflegroup
2193   \let\restituegroupe=\insertgroup
2194   \let\alafin=\lastchoices
2195   \let\formulaire=\AMCform
2196   \let\AMCdebutFormulaire=\AMCformBegin
2197   \let\champnom=\namefield
2198   \let\choixIntervalles=\AMCIntervals

```

## 5 Outputs

In the .xy file,  $1/\langle n \rangle$  means student sheet number 1 (there is only one “student sheet” for this document as we did not use `\onecopy`) and page number  $\langle n \rangle$  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:nom}{0sp}{37694935sp}{square}
\tracepos{0/34:nom}{6038827sp}{0sp}{square}
\tracepos{0/34:nom}{16026323sp}{0sp}{square}
\tracepos{0/34:nom}{0sp}{34709757sp}{square}

```

### 5.2 AMCboxedchar command

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

```
\tracepos{0/34:test}{22597209sp}{21638262sp}{square}
\tracepos{0/34:test}{23302629sp}{20932842sp}{square}
```

### 5.3 AMCcode command

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

```
\tracepos{0/57:case:code[5]:16,1}{25579605sp}{40511471sp}{square}
\tracepos{0/57:case:code[5]:16,1}{26285025sp}{39806051sp}{square}
\boxchar{0/57:case:code[5]:16,1}{A}
\tracepos{0/57:case:code[5]:16,2}{25579605sp}{39397359sp}{square}
\tracepos{0/57:case:code[5]:16,2}{26285025sp}{38691939sp}{square}
\boxchar{0/57:case:code[5]:16,2}{B}
\tracepos{0/57:case:code[5]:16,3}{25579605sp}{38283247sp}{square}
\tracepos{0/57:case:code[5]:16,3}{26285025sp}{37577827sp}{square}
\boxchar{0/57:case:code[5]:16,3}{C}
\tracepos{0/57:case:code[5]:16,4}{25579605sp}{37169135sp}{square}
\tracepos{0/57:case:code[5]:16,4}{26285025sp}{36463715sp}{square}
\boxchar{0/57:case:code[5]:16,4}{D}
\tracepos{0/57:case:code[4]:17,1}{27244404sp}{42739695sp}{square}
\tracepos{0/57:case:code[4]:17,1}{27949824sp}{42034275sp}{square}
\boxchar{0/57:case:code[4]:17,1}{0}
\tracepos{0/57:case:code[4]:17,2}{27244404sp}{41625583sp}{square}
\tracepos{0/57:case:code[4]:17,2}{27949824sp}{40920163sp}{square}
\boxchar{0/57:case:code[4]:17,2}{1}
\tracepos{0/57:case:code[4]:17,3}{27244404sp}{40511471sp}{square}
\tracepos{0/57:case:code[4]:17,3}{27949824sp}{39806051sp}{square}
\boxchar{0/57:case:code[4]:17,3}{2}
\tracepos{0/57:case:code[4]:17,4}{27244404sp}{39397359sp}{square}
\tracepos{0/57:case:code[4]:17,4}{27949824sp}{38691939sp}{square}
\boxchar{0/57:case:code[4]:17,4}{3}
\tracepos{0/57:case:code[4]:17,5}{27244404sp}{38283247sp}{square}
\tracepos{0/57:case:code[4]:17,5}{27949824sp}{37577827sp}{square}
\boxchar{0/57:case:code[4]:17,5}{4}
\tracepos{0/57:case:code[4]:17,6}{27244404sp}{37169135sp}{square}
\tracepos{0/57:case:code[4]:17,6}{27949824sp}{36463715sp}{square}
\boxchar{0/57:case:code[4]:17,6}{5}
\tracepos{0/57:case:code[3]:18,1}{28736261sp}{42739695sp}{square}
\tracepos{0/57:case:code[3]:18,1}{29441681sp}{42034275sp}{square}
\boxchar{0/57:case:code[3]:18,1}{0}
\tracepos{0/57:case:code[3]:18,2}{28736261sp}{41625583sp}{square}
\tracepos{0/57:case:code[3]:18,2}{29441681sp}{40920163sp}{square}
\boxchar{0/57:case:code[3]:18,2}{1}
\tracepos{0/57:case:code[3]:18,3}{28736261sp}{40511471sp}{square}
```

```

\tracepos{0/57:case:code[3]:18,3}{29441681sp}{39806051sp}{square}
\boxchar{0/57:case:code[3]:18,3}{2}
\tracepos{0/57:case:code[3]:18,4}{28736261sp}{39397359sp}{square}
\tracepos{0/57:case:code[3]:18,4}{29441681sp}{38691939sp}{square}
\boxchar{0/57:case:code[3]:18,4}{3}
\tracepos{0/57:case:code[3]:18,5}{28736261sp}{38283247sp}{square}
\tracepos{0/57:case:code[3]:18,5}{29441681sp}{37577827sp}{square}
\boxchar{0/57:case:code[3]:18,5}{4}
\tracepos{0/57:case:code[3]:18,6}{28736261sp}{37169135sp}{square}
\tracepos{0/57:case:code[3]:18,6}{29441681sp}{36463715sp}{square}
\boxchar{0/57:case:code[3]:18,6}{5}
\tracepos{0/57:case:code[2]:19,1}{30228118sp}{42739695sp}{square}
\tracepos{0/57:case:code[2]:19,1}{30933538sp}{42034275sp}{square}
\boxchar{0/57:case:code[2]:19,1}{0}
\tracepos{0/57:case:code[2]:19,2}{30228118sp}{41625583sp}{square}
\tracepos{0/57:case:code[2]:19,2}{30933538sp}{40920163sp}{square}
\boxchar{0/57:case:code[2]:19,2}{1}
\tracepos{0/57:case:code[2]:19,3}{30228118sp}{40511471sp}{square}
\tracepos{0/57:case:code[2]:19,3}{30933538sp}{39806051sp}{square}
\boxchar{0/57:case:code[2]:19,3}{2}
\tracepos{0/57:case:code[2]:19,4}{30228118sp}{39397359sp}{square}
\tracepos{0/57:case:code[2]:19,4}{30933538sp}{38691939sp}{square}
\boxchar{0/57:case:code[2]:19,4}{3}
\tracepos{0/57:case:code[2]:19,5}{30228118sp}{38283247sp}{square}
\tracepos{0/57:case:code[2]:19,5}{30933538sp}{37577827sp}{square}
\boxchar{0/57:case:code[2]:19,5}{4}
\tracepos{0/57:case:code[2]:19,6}{30228118sp}{37169135sp}{square}
\tracepos{0/57:case:code[2]:19,6}{30933538sp}{36463715sp}{square}
\boxchar{0/57:case:code[2]:19,6}{5}
\tracepos{0/57:case:code[1]:20,1}{31719975sp}{42739695sp}{square}
\tracepos{0/57:case:code[1]:20,1}{32425395sp}{42034275sp}{square}
\boxchar{0/57:case:code[1]:20,1}{0}
\tracepos{0/57:case:code[1]:20,2}{31719975sp}{41625583sp}{square}
\tracepos{0/57:case:code[1]:20,2}{32425395sp}{40920163sp}{square}
\boxchar{0/57:case:code[1]:20,2}{1}
\tracepos{0/57:case:code[1]:20,3}{31719975sp}{40511471sp}{square}
\tracepos{0/57:case:code[1]:20,3}{32425395sp}{39806051sp}{square}
\boxchar{0/57:case:code[1]:20,3}{2}
\tracepos{0/57:case:code[1]:20,4}{31719975sp}{39397359sp}{square}
\tracepos{0/57:case:code[1]:20,4}{32425395sp}{38691939sp}{square}
\boxchar{0/57:case:code[1]:20,4}{3}
\tracepos{0/57:case:code[1]:20,5}{31719975sp}{38283247sp}{square}
\tracepos{0/57:case:code[1]:20,5}{32425395sp}{37577827sp}{square}
\boxchar{0/57:case:code[1]:20,5}{4}
\tracepos{0/57:case:code[1]:20,6}{31719975sp}{37169135sp}{square}
\tracepos{0/57:case:code[1]:20,6}{32425395sp}{36463715sp}{square}

```

```
\boxchar{0/57:case:code[1]:20,6}{5}
```

# 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>11</b>
3.1	Package options . . . . .	11
3.2	Questions and answers . . . . .	12
3.3	Scoring . . . . .	14
3.4	Groups of questions . . . . .	14
3.5	Students identification . . . . .	16
3.6	Separate answer sheet . . . . .	17
3.7	Random computation questions . . . . .	17
3.8	French command names . . . . .	20
3.9	Customisation . . . . .	20
3.9.1	Boxes . . . . .	20
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	Dutch . . . . .	27
4.4.3	French . . . . .	27
4.4.4	German . . . . .	28
4.4.5	Italian . . . . .	28
4.4.6	Norwegian . . . . .	28
4.4.7	Portuguese . . . . .	29
4.4.8	Spanish . . . . .	29
4.4.9	Japanese . . . . .	29
4.4.10	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 . . . . .	30
4.6.3	Tokens shuffling . . . . .	31
4.7	Keys numbering . . . . .	31
4.8	Boxes . . . . .	32
4.8.1	Position logging . . . . .	32



4.8.2	Boxes to be checked by students . . . . .	34
4.8.3	Scoring zones . . . . .	40
4.8.4	Binary boxes . . . . .	40
4.9	Checking Environment . . . . .	41
4.10	Handling groups of questions . . . . .	41
4.11	Questions . . . . .	45
4.11.1	Managing answers . . . . .	46
4.11.2	Separate answer sheet . . . . .	47
4.11.3	Formatting answers . . . . .	51
4.11.4	Score zones . . . . .	52
4.11.5	Formatting questions . . . . .	54
4.11.6	Explanations . . . . .	56
4.12	Scoring . . . . .	56
4.13	Numerical data . . . . .	57
4.13.1	Codes . . . . .	57
4.13.2	Numerical questions . . . . .	60
4.13.3	Intervals . . . . .	71
4.14	Open questions . . . . .	72
4.15	Boxes with letters only . . . . .	75
4.16	Page formatting . . . . .	76
4.16.1	Watermark . . . . .	76
4.16.2	Signs for scan analysis . . . . .	77
4.17	Defining a single exam copy content . . . . .	78
4.18	Pre-association . . . . .	80
4.19	Package options . . . . .	80
4.20	Package Errors . . . . .	81
4.21	Optional features . . . . .	81
4.22	External control . . . . .	82
4.23	Page layout . . . . .	83
4.24	Initialisation . . . . .	83
4.25	French command names . . . . .	84
<b>5</b>	<b>Outputs</b>	<b>84</b>
5.1	namefield command . . . . .	84
5.2	AMCboxedchar command . . . . .	84
5.3	AMCcode command . . . . .	85

# 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.

Symbols	
\"	119, 125
\@aucune	759, 763, 764
\@firstoftwo	585
\@ifstar	922, 923
\@secondoftwo	587
\@skiphyperreffalse	1076
\@skiphyperreftrue	1072, 1076
\@tempboxa	1903, 1907, 1908
\@tempdimb	1892, 1894
\@tempdimc	1893, 1894
\~	157, 158, 163, 164
\_	1097
A	
\aa	144
\alafin	21, 2194
\amc	786, 797, 799, 801, 802, 804, 807, 808, 810, 813, 814, 817–820, 822, 824, 825, 827, 829, 830, 832, 834, 835, 838, 839, 841, 842, 844, 845, 847, 848, 850, 851, 853, 856, 857, 859, 861, 865–868, 1135–1139, 1141, 1148, 1151, 1156, 1160, 1168, 1171–1173, 1178, 1180, 1182– 1185, 1187, 1192, 1194, 1196, 1204, 1209–1211, 1213, 1215, 1218–1221, 1223, 1225, 1228, 1229, 1231, 1232, 1234–1237, 1257, 1258, 1263– 1268, 1273, 1275– 1279, 1281, 1283– 1286, 1289, 1296–1298, 1300, 1302, 1303, 1305, 1307, 1312, 1313, 1315, 1317–1319, 1321, 1349, 1351, 1353, 1355, 1357, 1360, 1363, 1366, 1369, 1372, 1375, 1378, 1381, 1384, 1391, 1393, 1397, 1401, 1403, 1415, 1417, 1419, 1422, 1424, 1427, 1430, 1433, 1437, 1439, 1441, 1448, 1450, 1453, 1462–1465, 1467, 1469, 1472, 1477, 1478, 1480, 1484, 1486, 1489– 1492, 1494–1496, 1499, 1502, 1507, 1509, 1512, 1516–1518, 1520, 1523, 1528, 1534, 1541, 1542, 1549, 1551–1553, 1555, 1558, 1560, 1563, 1566, 1567, 1572, 1582– 1588, 1590, 1595, 1596, 1599, 1600, 1603, 1604, 1606–1610, 1612, 1613, 1616, 1618, 1621–1625, 1627, 1629, 1631, 1634, 1639–1646, 1648–1650, 1652, 1653, 1660, 1662–1665, 1669–1671, 1675–1678, 1680–1682, 1694, 1696, 1702, 1711, 1714, 1715, 1721– 1727, 1729, 1730, 1733 \AMC@addpages 2007, 2009, 2011 \AMC@affiche . . . . . <u>245</u> , 1089 \AMC@amclog 8, 246, 545, 911, 955, 960, 1107, 1126– 1129, 1158, 1248, 1405, 1426, 1428, 1432, 1434, 1442, 1455, 1471, 1554, 1557, 1573, 1575, 1592, 1628, 1644, 1648, 1651, 1655, 1689, 1786, 1842, 1861, 1878, 1984, 1998, 2084, 2088, 2160–2163 \AMC@answerBox . . . . . <u>432</u> \AMC@answerBox@ . . . . . . . . . . <u>318</u> , 440, 486, 488, 504, 506, 561, 563 \AMC@binaryBoxes . . . . . . . . . . <u>555</u> , 1921–1923 \AMC@box . . . <u>473</u> , 957, 958, 961 \AMC@boxcolor . 336, 457, 1815 \AMC@boxcolor@ . . . . . . . . . . 336–338, 340, 355, 364, 376, 380, 395 \AMC@boxeddown . 347, 445, 456 \AMC@boxedheight . . . . . . . . . . 360, 374, 375, 377, 378, 381, 382, 424, 447, 451, 452, 463, 466 \AMC@boxedrule . . . . . . . 352, 374–376, 443, 454 \AMC@boxedwidth . . . . . . . . . . 363, 374, 375, 377, 378, 381, 382, 424, 446, 451, 453, 463, 464 \AMC@catalogMode . . . . . . . . . . 2059, 2064, 2138 \AMC@CBtaille 553, 1945, 2069 \AMC@checkbox . . . . 318, 339, 342, 388, 390, 399, 405, 415, 561, 957, 1406, 1417, 1427, 1430 \AMC@chiffres . . . . . 1130 \AMC@crosschar . . . . 358, 459 \AMC@crossrule . 380, 444, 460 \AMC@definitnumero . . 246, 250 \AMC@draw@crossfalse . . . 341 \AMC@draw@crosstrue . . . . 343 \AMC@error@explain . . . . . . . . . . 1121, 1123, <u>2101</u> \AMC@fillcolor@ . . . . . . . 339, 354, 355, 371, 376 \AMC@fin@rep <u>746</u> , 930, 934, 938 \AMC@formBox . . . . . <u>473</u> \AMC@formBox@ . . . . . <u>473</u> \AMC@fullGroupsfalse . . . . 25 \AMC@fullGroupstrue 2039, 2062 \AMC@if@separate@question . . . . . 797,

872, 877, 884, 890, 1242, 1706, 1850, 1886	\AMC@LR ..... 12, 346, 1895, 1940–1944, 2153	\AMC@prepare ... 248, 251, 252
\AMC@imax ..... 639, 651–653	\AMC@makeovalbox 369, 387–390	\AMC@prepare@element ... ..... 602, 610, 720
\AMC@intituleHead ..... ..... 1917, 1934, 2063	\AMC@mem@add ..... 827, 873, 880, 886, 893, 1243	\AMC@printformoutside@false ..... 477
\AMC@keepmemoryfalse .. 1999	\AMC@mem@add@ifneeded .. ..... 784, 1103	\AMC@printformoutside@true ..... 479, 480
\AMC@keepmemorytrue .... 912	\AMC@mem@addsingle@ifneeded .. 876, 915, 917, 919, 921	\AMC@printkeyoutside@false ..... 494
\AMC@keyBox@ ..... ... 502, 1159, 1408, 1411	\AMC@mem@addvar ..... 832	\AMC@printkeyoutside@true ..... 496, 498
\AMC@lang@code ..... .. 2036, 2065, 2066, 2174	\AMC@mem@aid ..... 851, 885	\AMC@qaff 769, 1069, 1097, 1117
\AMC@loc@catalog ..... 85, 98, 110, 124, 137, 149, 160, 173, 184, 2063	\AMC@mem@answer ..... .. 883, 939, 943, 948, 950	\AMC@save@box . 320, 372, 2093
\AMC@loc@corrected ..... 84, 97, 109, 123, 136, 148, 159, 172, 183, 1917	\AMC@mem@category ..... 845, 879, 1706, 1850, 1886	\AMC@setcolors@ ..... ..... 335, 353, 370, 393
\AMC@loc@DE ..... 117	\AMC@mem@clear .... 802, 2001	\AMC@shape@form ..... 427
\AMC@loc@draft ..... 79, 92, 104, 118, 131, 143, 155, 167, 179, 1890	\AMC@mem@next .. 822, 878, 891	\AMC@shape@form@base 414, 428
\AMC@loc@ES ..... 166	\AMC@mem@openQuestion .. ..... 889, 1102	\AMC@shape@form@ticked . ..... 416, 418, 421
\AMC@loc@explain .... 86, 111, 125, 161, 185, 1121	\AMC@mem@qidaffname . 839, 892	\AMC@shape@none ..... 431
\AMC@loc@FR .. 103, 2065, 2189	\AMC@mem@show .. 866, 902, 911	\AMC@shape@oval ..... 392
\AMC@loc@IT ..... 130	\AMC@mem@show@filter 867, 906	\AMC@shape@square ..... 351
\AMC@loc@JA ..... 178	\AMC@mn@leftmargin ..... ..... 967, 975, 978, 984	\AMC@shapename ..... .. 253, 261, 270, 279, 450
\AMC@loc@message ..... 80, 93, 105, 119, 132, 144, 156, 168, 180, 2153	\AMC@mn@rightmargin .... ..... 968, 976, 979, 982	\AMC@shapename@ 253, 347, 468
\AMC@loc@namesurname 90, 115	\AMC@mn@sep .... 966, 982, 984	\AMC@shapeprepare@form . 413
\AMC@loc@NL ..... 91	\AMC@mn@test ... 965, 972, 981	\AMC@shapeprepare@none . 430
\AMC@loc@NO ..... 142	\AMC@NCBcheck 549, 1923, 2069	\AMC@shapeprepare@oval . 386
\AMC@loc@none ..... 87, 99, 112, 126, 138, 150, 162, 174, 186, 759	\AMC@NCBetud . 549, 1921, 2068	\AMC@shapeprepare@square 350
\AMC@loc@PT ..... 154	\AMC@NCBpage . 549, 1922, 2068	\AMC@shuffletoks ..... ..... 226, 672, 675, 739
\AMC@loc@q ..... 83, 96, 108, 122, 135, 147, 158, 171, 182, 1079	\AMC@new@savebox ..... ... 319, 2092, 2096–2099	\AMC@smashbox 323, 326–330, 333
\AMC@loc@qf ..... 82, 95, 107, 121, 134, 146, 157, 170, 181, 768	\AMC@note ..... 1918, 1919, 1953, 1963, 2152	\AMC@smashboxheight .... ..... 324, 327–329, 332
\AMC@loc@question ..... 88, 100, 113, 127, 139, 151, 163, 175, 187, 193	\AMC@numeric@scoreapprox ..... 1389, 1635	\AMC@smashcentered ..... .. 325, 364, 365, 395, 396
\AMC@loc@questions ..... 89, 101, 114, 128, 140, 152, 164, 176, 188, 193	\AMC@numeric@scoreexact ..... 1387, 1630	\AMC@SR ... 195, 197, 200– 203, 205, 207, 208, 1993
\AMC@logfile ..... 8–10	\AMC@numerotation 245, 249, 250	\AMC@SR@count 200, 201, 203, 211, 215, 218, 221–224
	\AMC@outside@sep ... 449, 455	\AMC@SR@time ..... 212, 2050
	\AMC@oval@radius ..... ..... 376, 448, 464, 466	\AMC@SRadvance . 198, 205, 206
	\AMC@ovalbox@ . 389, 402, 2098	\AMC@SRbit ..... 205
	\AMC@ovalbox@F 390, 400, 2099	\AMC@SRconst ... 196, 200, 203
	\AMC@ovalbox@R 387, 408, 2096	\AMC@SRmax ..... 211, 239
	\AMC@ovalbox@RF 388, 406, 2097	\AMC@SRnextByte ..... 211
	\AMC@pagepos ..... 253, 1930	\AMC@SRnum . 213, 214, 216, 217, 221, 224, 239, 240
	\AMC@premierecopie ..... ... 554, 1986, 1989, 2070	\AMC@SRset . 197, 209, 210, 212
		\AMC@SRtest ..... 206, 217

\AMC@SRvalue . . . . .	208	\AMC@use@box . . . . .	321,	\AMCcodeH . . . . .	1237
\AMC@stepQuestion . . . . .		400, 402, 406, 408, 2094		\AMCcodeHspace 1131, 1163, 1198	
. . . . .	1069, 1088, 1114	\AMC@VERSION . . . . .	2167, 2169	\AMCcodeVspace . . . . .	
\AMC@sti 226, 234, 238, 241, 242		\AMC@watertext . . . . .	1890, 1901	. . . . .	1132, 1161, 1197, 1200
\AMC@stil 227, 235–237, 239, 243		\AMC@XYFILE . . . . .	256, 265,	\AMCcompleteMulti . . . . .	13, 50
\AMC@sz@box . . . . .		274, 286, 290, 291, 295,		\AMCcurrentenv 581, 584, 1087	
. . . . .	988, 990, 992, 995, 1038	304–306, 1992, 2032,		\AMCdebutFormulaire . . . . .	21, 2196
\AMC@sz@callin . . . . .		2165–2174, 2176, 2178		\AMCdecimalPoint . . . . .	
. . . . .	1000, 1005, 1013,	\AMC@XYspecial . . . . .		. . . . .	1337, 1481, 1513
1015, 1080, 1098, 1099		. . . . .	300, 303, 305, 306	\AMCdefault@groupmode . . . . .	
\AMC@sz@callin@question 990		\AMCaddpagesto . . . . .	2007	. . . . .	599, 613, 614
\AMC@sz@callout 999, 1004,		\AMCanswer 950, 951, 1804, 1865		\AMCdontAnnotate . . . . .	290, 1819
1008, 1010, 1092, 1093		\AMCassociation . . . . .	2031	\AMCdontScan . . . . .	290, 1818
\AMC@sz@callout@margin . . . . .	992	\AMCbeforeQuestion . . . . .		\AMCdum@reponses . . . . .	739, 753
\AMC@sz@callout@margin . . . . .	995	. . . . .	1078, 1091, 1143	\AMCemptybox . . . . .	963, 988, 1023
\AMC@sz@depth . . . . .	988, 998, 1003	\AMCbeginAnswer . . . . .		\AMCendAnswer . . . . .	
\AMC@sz@height 988, 997, 1002		. . . . .	937, 951, 1803, 1864	. . . . .	938, 951, 1803, 1864
\AMC@sz@init@margin . . . . .	994	\AMCbeginQuestion . . . . .		\AMCexponent . . . . .	1339, 1667, 1679
\AMC@sz@width . . . . .	988, 996, 1001	. . . . .	1078, 1097, 1117, 1142	\AMCform . . . . .	17, 784, 2195
\AMC@sza@box . . . . .	1023, 1031, 1034	\AMCbin@begin . . . . .	564, 1921–1923	\AMCformAfterQuestion . . . . .	
\AMC@sza@callin 778, 779,		\AMCbin@digit . . . . .	559–564	. . . . .	767, 1103
1045, 1050, 1058, 1060		\AMCbin@id . . . . .	558, 561, 563, 564	\AMCformAnswer . . . . .	766
\AMC@sza@callin@margin 1035		\AMCbin@ndigits . . . . .		\AMCformAnswerA . . . . .	783, 886
\AMC@sza@callin@margin 1039		. . . . .	557, 567, 570, 576, 579	\AMCformatChoices . . . . .	
\AMC@sza@callin@none . . . . .	1027	\AMCbin@number . . . . .		. . . . .	1240, 1707, 1851, 1887
\AMC@sza@callin@question 1031		. . . . .	556, 566, 569, 571, 573	\AMCformBeforeQuestion . . . . .	
\AMC@sza@callout 774, 775,		\AMCbin@one . . . . .	560, 571	. . . . .	766, 772
1044, 1049, 1053, 1055		\AMCbin@sequence . . . . .	555,	\AMCformBegin . . . . .	17, 784, 2196
\AMC@sza@callout@margin 1034		566, 571, 572, 577, 578		\AMCformFilter . . . . .	904
\AMC@sza@callout@margin 1038		\AMCbin@zero . . . . .	562, 572, 577	\AMCformHSpace . . . . .	53, 782, 1865
\AMC@sza@callout@none . . . . .	1026	\AMCbloc . . . . .	1085	\AMCformQuestion . . . . .	766
\AMC@sza@callout@question . . . . .	1030	\AMCbo@help . . . . .	1854, 1855, 1868	\AMCformQuestionA . . . . .	770, 893
\AMC@sza@depth 1023, 1043, 1048		\AMCboHide . . . . .	1881, 1887	\AMCformQuestionN . . . . .	769, 777
\AMC@sza@height . . . . .		\AMCboOpts . . . . .	1858	\AMCformS . . . . .	784
. . . . .	1023, 1042, 1047	\AMCboShow . . . . .	1859, 1887	\AMCformVSpace . . . . .	53, 766
\AMC@sza@init@margin . . . . .	1033	\AMCbotextGoto . . . . .	1853, 1882	\AMCgroup@pre . . . . .	668, 691
\AMC@sza@init@margin . . . . .	1037	\AMCboxColor . . . . .	471, 1945	\AMCgrouploop@next . . . . .	
\AMC@sza@init@none . . . . .	1025	\AMCboxDimensions . . . . .	472, 566	. . . . .	693, 704, 719
\AMC@sza@init@question 1029		\AMCBoxedAnswers . . . . .	924	\AMCgrouploop@prep . . . . .	
\AMC@sza@width 1023, 1041, 1046		\AMCBoxOnly . . . . .	1853	. . . . .	679, 702, 717
\AMC@tempenv . . . . .	583, 584	\AMCboxOutsideLetter . . . . .		\AMCgrouppre@cyclic . . . . .	662
\AMC@tmpXY . . . . .	301, 304, 306	. . . . .	473, 1805, 1866	\AMCgrouppre@fixed . . . . .	624
\AMC@tracebox 253, 307, 394,		\AMCboxStyle . . . . .	20, 443, 2085	\AMCgrouppre@withoutreplacement . . . . .	640
420, 537, 539, 542, 1913		\AMCcercle . . . . .	1909, 1912	. . . . .	631
\AMC@tracechar . . . . .	293, 490, 508	\AMCchoiceLabel . . . . .	432, 512	\AMCgrouppre@withreplacement . . . . .	
\AMC@tracepos . . . . .	254, 361, 367	\AMCchoiceLabelFormat 348, 432		. . . . .	944, 945, 949
\AMC@traceposx . . . . .	263, 283	\AMCcleardoublepage . . . . .	17, 2016	\AMChorizAnswerSep . . . . .	
\AMC@traceposy . . . . .	272, 282, 284	\AMCcode . . . . .	1236	. . . . .	946, 947, 949
\AMC@unnumero . . . . .	245	\AMCcodeGrid . . . . .	16, 1130	\AMChorizBoxSep . . . . .	
		\AMCcodeGridInt . . . . .	16, 1130		

\AMCid@check . . . . .	\AMCload@counter . . . . .	\AMCocol@FrameRule . . . . .
. 18, 288, 1923, 1928,	. . . 16, 728–731, 734,	. . . . . 1740, 1764, 1835
1929, 1946, 1950, 2159	735, 739, 742–745, 748	\AMCOpen . . . . . 1736
\AMCid@checkmax . . . . .	\AMCload@reponse . . . 733, 752	\AMCopen@answer . . . . .
. . . 549, 1929, 2068, 2159	\AMClocalized . . . . . 78	. . . 1742, 1749, 1777, 1793
\AMCid@etud . . . . .	\AMCloop@k . . . . 645, 651,	\AMCopen@boxframerule . .
. . . 19, 258, 267, 276,	655, 657, 678, 686, 695	. . . . . 1761, 1762, 1808
288, 290, 291, 296, 312,	\AMCmarginNote 969, 995, 1038	\AMCopen@boxmargin . . . . .
428, 1920, 1921, 1946,	\AMCmem@elt@cat 842, 854, 858	. . . . . 1759, 1760, 1807
1950, 1958, 1985, 1986,	\AMCmessage . . . . 8, 1090, 1103	\AMCopen@contentcommand
1988, 1993, 1998, 2003,	\AMCncol@Background 1347, 1659	. . . . . 1750, 1836
2005, 2023, 2024, 2033	\AMCncol@Border . . 1343, 1659	\AMCopen@framerule . . . . .
\AMCid@etudfin . . . . .	\AMCncol@BorderWidth . .	. . . . . 1765, 1766, 1834
. . . . 21, 1987–1989, 2003	. . . . . 1345, 1658	\AMCopen@Hspace . . . 1751,
\AMCid@etudstart 20, 1985, 2005	\AMCneeds@storeboxfalse 2075	1752, 1806, 1811, 1822
\AMCid@name . . . . .	\AMCneeds@storeboxtrue 2076	\AMCopen@LineHeight . . . . .
52, 518, 521, 523, 527,	\AMCnobloc . . . . . 1084, 1147	. . . . . 1755, 1756, 1776
529, 836, 892, 1089,	\AMCnoCompleteMulti . . . . .	\AMCopen@Lines 1757, 1758, 1781
1159, 1408, 1411, 1748	. . . . . 13, 51, 1802	\AMCopen@lines . . . . . 1774
\AMCid@quest . 17, 290, 291,	\AMCNombreCopies . . . . .	\AMCopen@lineuptext 1744,
518, 521, 523, 527, 529,	. . . 1982, 1983, 2175, 2178	1767, 1791, 1794, 1797
537, 539, 542, 835, 892,	\AMCnoScoreZone . . 1068, 1144	\AMCopen@question . . . . .
1089, 1090, 1159, 1409,	\AMCnontextGoto . . . 1328, 1698	. . . 1743, 1748, 1810, 1811
1412, 1592, 1786, 1861	\AMCnontextSign 1335, 1532, 1533	\AMCopen@Width 1753, 1754, 1775
\AMCIDBoxesA . . . . 1921, 1925	\AMCnontextVHead . . 1329, 1485	\AMCopenHide . . . . 1845, 1851
\AMCIDBoxesABC . . . 1924, 1947	\AMCnum@copies . . . . .	\AMCopenOpts . . . . . 1773
\AMCIDBoxesB . . . . 1922, 1926	. . . . 22, 1982–1984, 1987	\AMCopenShow . . . . 1784, 1851
\AMCIDBoxesC . . . . 1923, 1926	\AMCnum@questions . . 740, 742	\AMCotextGoto . . . . 1736, 1846
\AMCids@height . . 63, 74, 1946	\AMCnumeric@Hspace . . . . .	\AMCotextReserved . 1737,
\AMCids@sidefalse . . . 66, 68	. . . 1330, 1342, 1443,	1747, 1821, 1825, 1827
\AMCids@sidetrue . . . . . 70	1445, 1481, 1500, 1679	\AMCoutsideLabelFormat . 473
\AMCids@topfalse . . . . 66, 70	\AMCnumeric@Vspace . . . . .	\AMCpageref . . . . . 2027
\AMCids@toptrue . . . . . 68	. . . 1331, 1341, 1425,	\AMCpostOquest . . . . 59, 1838
\AMCids@width . . . 62, 73, 1949	1431, 1457, 1473, 1487,	\AMCqlabel . . . . . 2028
\AMCidsPosition . . . . . 60	1521, 1533, 1666, 1668	\AMCquestionaff . . . . . 1069
\AMCidsVar . . . . . 64	\AMCnumericChoices . . 18, 1240	\AMCquestionNumberfalse 1145
\AMCidsVarN . . . . . 64, 65	\AMCnumericChoicesPlain	\AMCquestionNumbertrue . . 33
\AMCif@env . . 582, 1121, 1123	. . . . . 1705, 1712	\AMCrandomseed . . . . . 209
\AMCifcategory . . . . . 868	\AMCnumericHide . . 1702, 1707	\AMCref . . . . . 2024
\AMCinterBquest . . . . 58, 1103	\AMCnumericOpts . . . . 1401	\AMCrep@@count . 754, 756, 758
\AMCinterBrep . . . . 53, 942	\AMCnumericShow . . 1694, 1707	\AMCrep@bloc . . . . . 927, 939
\AMCinterIquest . . . . 57, 1103	\AMCnumero . . . . . 1070, 1997	\AMCrep@count . . . . . 518,
\AMCinterIrep . . . . 53, 928	\AMCocol@Background . . . . .	521, 523, 527, 529, 727,
\AMCIntervalFormat 1718, 1729	. . . . . 1738, 1745, 1809	736, 756, 758, 762, 848,
\AMCIntervals . 17, 1718, 2198	\AMCocol@BoxFrameRule . .	885, 926, 932, 936, 954,
\AMClabel . . . . . 2024, 2029	. . . . . 1739, 1763, 1809	955, 958–961, 1153,
\AMCload@reponse . . . . .	\AMCocol@Foreground . . . . .	1157–1159, 1241, 1243,
. . . . . 750, 752, 957, 961	. . . . . 1741, 1746,	1404, 1405, 1409, 1412
	1814, 1815, 1824, 1827	\AMCrep@fini 750, 753, 757, 765
		\AMCrep@init 746, 929, 932, 936



questionouverte ... <a href="#">1083</a>	1106, 1126–1129, 1158,	\ifAMCformulaire@dedans
reponses ..... <a href="#">21</a>	1992, 1998, 2032, 2164	..... 43, 479,
reponseshoriz ..... <a href="#">21</a>	\ifAMC@correc ..... 28,	520, 536, 1591, 1688,
reponsesperso ..... <a href="#">21</a>	338, 957, 1407, 1777, 1793	1785, 1841, 1860, 1877
\evensidemargin .... 976, 978	\ifAMC@correchead .....	\ifAMCids@side ..... 61, 1949
\examcopy ..... 2191	..... 26, 1120, 1931,	\ifAMCids@top ..... 60, 1946
examcopy (environment) . <a href="#">2112</a>	1974, 1989, 2145, 2151	\ifAMCneeds@storebox ...
\exemplaire ..... <a href="#">21</a> , 2189	\ifAMC@draw@cross 322, 358, 379	..... 2075, 2080, 2091
\exemplairepair ..... <a href="#">2023</a>	\ifAMC@ensemble . 41, 478,	\ifAMCquestionNumber 33, 1076
\explain ..... <a href="#">1119</a> , 2102	495, 516, 535, 787, 898,	\ifAMCsz@logged .... 533, 544
\ExplSyntaxOff . 870, 1239,	901, 905, 910, 1097,	\ifAMCtype@multi ... 37, 761
1327, 1704, 1717, 1735	1246, 1591, 1687, 1688,	\ifAMCune@bonne .... 36, 763
\ExplSyntaxOn . 784, 1133,	1785, 1790, 1804, 1810,	\ifcase ..... 65
1253, 1332, 1709, 1719	1841, 1860, 1865, 1868,	\ifcsname .... 1007, 1012,
	1877, 1996, 2160, 2170	1017, 1052, 1057, 1062
<b>F</b>	\ifAMC@fullGroups .. 25, 681	\ifdim ..... 981
\fancypagestyle .....	\ifAMC@inside@box .....	\ifKV@AMCBoxOnly@ordered 1869
.. 1933, 1938, 1960, 1967	..... 39, 498, 526,	\ifKV@AMCdim@cross .....
\fbox ..... 1958	1805, 1866, 2161, 2171	..... 342, 354, 371
\footrulewidth .....	\ifAMC@inside@digit . 42, 433	\ifKV@AMCOpen@annotate 1819
.. 1936, 1955, 1965, 1970	\ifAMC@invisible ... 48, 253	\ifKV@AMCOpen@dots .... 1778
\formulaire ..... <a href="#">21</a> , 2195	\ifAMC@keepmemory . 908, 2001	\ifKV@AMCOpen@lineup ...
\fp .... 1267, 1275, 1277,	\ifAMC@ordre ..... 23, 747	..... 1789, 1831
1279, 1283, 1288, 1290,	\ifAMC@outside@box .....	\ifKV@AMCOpen@scan .... 1818
1292, 1296, 1303, 1308,	. 40, 478, 496, 2162, 2172	indivanswers (option) .. 7, 11
1315, 1582, 1609, 1648,	\ifAMC@pagelayout .....	init (option) ..... 11
1664, 1677, 1721, 1722,	... 45, 1928, 1973, 2144	\insertgroup ... 15, <a href="#">671</a> , 2193
1724–1727, 1729, 1730	\ifAMC@pdfform .....	\insertgroupfrom ... 15, <a href="#">671</a>
francais (option) ..... 11	49, 310, 1990, 2004, 2083	insidebox (option) ..... 11
fullgroups (option) ..... 12	\ifAMC@plain ..... 35, 2104	\int 799, 801, 805, 811, 818,
<b>G</b>	\ifAMC@postcorrect .....	857, 1137, 1148, 1182,
\group ..... 1259, 1271	.... 46, 764, 2163, 2173	1204, 1268, 1284, 1312,
<b>H</b>	\ifAMC@printformoutside	1315, 1317–1319, 1321,
\he@dbas .... 1915, 1940, 1941	..... 476, 485	1416, 1438, 1449, 1456,
\he@dhaut ..... 1916, 1944	\ifAMC@printformoutside@	1462, 1464, 1466, 1472,
\he@dtaille ..... 1914–1916	..... 475, 482	1478, 1480, 1485, 1491,
\hfuzz ..... 329	\ifAMC@printkeyoutside .	1495, 1499, 1502, 1509,
\ht ..... 327, 964	..... 493, 503	1511, 1517, 1520, 1523,
<b>I</b>	\ifAMC@printkeyoutside@	1551, 1552, 1555, 1558,
\ifAMC@affichekeys .....	..... 492, 500	1560, 1563, 1566, 1567,
..... 27, 1096, 1097	\ifAMC@qbloc ..... 29,	1585, 1588, 1596, 1599,
\ifAMC@asqbloc .. 30, 767, 773	1078, 1095, 1103, 1116,	1600, 1606, 1608, 1612,
\ifAMC@automarks .....	1118, 1699, 1847, 1883	1623, 1629, 1631, 1634,
47, 898, 1996, 2012, 2019	\ifAMC@rbloc .... 31, 927, 930	1641, 1646, 1652, 1653,
\ifAMC@calibration 34, 255,	\ifAMC@shuffleG . 24, 672, 675	1665, 1671, 1678, 1682
264, 273, 286, 290, 291,	\ifAMC@watermark 38, 1931, 2150	
294, 955, 960, 1090,	\ifAMC@zoneformulaire ..	<b>K</b>
	..... 44, 480, 517, 788	\keys ..... 1167, 1179,
	\ifAMCcomplete@multi 32, 761	1334, 1398, 1594, 1697

<b>L</b>		
\lastchoices	.....	746, 2194
\linebreak	.....	1800
<b>M</b>		
\m@rque	.....	1913, 1940–1943
\m@rqueCalage	.....	1912, 1913
\marginpar	.....	992, 1034
\mauvaise	.....	21, 2185
\melangegroupe	.....	21, 2192
\MessageBreak	.....	2102
\multiSymbole	.....	1078, 1105, 1110
<b>N</b>		
\namefield	....	16, 307, 2197
\namefielddots	.....	308
\newbox	.....	323
\newsavebox	.....	319
\newstorebox	.....	2092
\nobreak	.....	474, 1832
nopage (option)	.....	6, 12
noshuffle (option)	.....	11
noshufflegroups (option)	..	11
\nouveaugroupe	.....	590, 712
nowatermark (option)	...	5, 11
\NoWatermarkExterne	...	2120
<b>O</b>		
\oddsidemargin	.....	975, 979
\onecopy	....	1981, 2116, 2189
options:		
answers	.....	11
asbox	.....	11
automarks	.....	12
bloc	.....	21
box	.....	11
calibration	.....	5, 11
catalog	.....	11
chiffres	.....	21
completemulti	.....	11
correc	.....	21
correcindiv	.....	21
digits	.....	11
ensemble	.....	21
français	.....	11
fullgroups	.....	12
indivanswers	.....	7, 11
init	.....	11
insidebox	.....	11
nopage	.....	6, 12
noshuffle	.....	11
noshufflegroups	.....	11
nowatermark	.....	5, 11
ordre	.....	21
outsidebox	.....	11
pdfform	.....	12
plain	.....	11
postcorrect	.....	12
separateanswersheet	..	5, 11, 17
storebox	.....	12
\or	.....	67, 69
ordre (option)	.....	21
outsidebox (option)	.....	11
\ouverte@vs	.....	1083
<b>P</b>		
\PackageError	.....	579, 617, 688, 2101
\pageref	.....	2027
pdfform (option)	.....	12
\pgfextractx	.....	972
\pgfpointanchor	.....	973
\pgfpointdiff	.....	972
\pgfpointorigin	.....	972
plain (option)	.....	11
postcorrect (option)	.....	12
\prg	.....	786, 789, 791, 794, 1220, 1316
<b>Q</b>		
\question	.....	1083
question (environment)	12, 1083	
\QuestionIndicative	.....	14, 1126, 1152
questionmult (environment)	.....	12, 1083
questionouverte (environment)	.....	1083
<b>R</b>		
\raisebox	.....	1338
\refstepcounter	.....	1076
\reponses	.....	2181
reponses (environment)	...	21
\reponseshoriz	.....	2182
reponseshoriz (environment)	.....	21
\reponsesperso	.....	2183
reponsesperso (environment)	.....	21
\restituegroupe	....	21, 2193
\rngstate	.....	1993
<b>S</b>		
\savebox	.....	320
\sbox	.....	964
\scoring	.....	14, 1126, 2186
\scoringDefaultM	14, 1126,	2187
\scoringDefaultS	14, 1126,	2188
\ScoringExterne	.....	2120
\section	.....	915, 919
separateanswersheet (option)	.....	5, 11, 17
\seq	.....	1136, 1184, 1192, 1258, 1266–1268
\setdefaultgroupmode	...	613
\setgroupmode	.....	599, 613
\shuffle@it	.....	739, 753
\shufflegroup	.....	15, 637, 649, 671, 2192
\shufflegroupslice	.....	654, 657, 674
\smash	.....	1777, 1793
\space	.....	2102
\storebox	.....	2093
storebox (option)	.....	12
\str	.....	854
\strut	.....	982, 984
\SujetExterne	.....	2120
<b>T</b>		
\textasciicircum	.....	1340
\TextField	.....	312
\textit	..	86, 111, 125, 161, 185
\textsc	.....	1934
\textsf	.....	459
\times	.....	1340
\tl	819, 820, 825, 830, 859, 861, 1255, 1257, 1261, 1264, 1265, 1550, 1561, 1562, 1565, 1569, 1570, 1577, 1586, 1587, 1645	
\tracepos	.....	257, 266, 275
<b>U</b>		
\une@rep	.....	926, 927, 932, 936, 957, 961
\useasboundingbox	.....	374
\usebox	.....	321
\usestorebox	.....	2094



	<b>V</b>		
<code>\version</code>	2167	<code>\vtop</code>	1187
<code>\vfuzz</code>	329		<code>\with</code> .. 2168–2174, 2176, 2178
<code>\vrule</code>	1537	<b>W</b>	
		<code>\wd</code>	329, 330, 964
		<code>\wrongchoice</code>	14, 763, 764, <u>954</u> , 1728, 2185