

The automultiplechoice package*

Alexis Bienvenue
paamc@passoire.fr

February 12, 2021

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 \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 \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:d3d7ea24` from AMC 1.5.0~rc2

```

        \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^AT_EX 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^AT_EX 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^AT_EX 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^AT_EX 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
---	---	---	---

E


Question 3:

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


Question 4:

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

DRAFT



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



~2/1/88~

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 scan. 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 1807?

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

Question 2 ▼ What is the capital of Iceland?

- ☐ A Dighouthi
- ☐ B Dhaka
- ☐ C Dublin
- ☐ D Dili


Question 3 ▲ What is the capital of Ghana?

- ☐ A Apia
- ☐ B Accra
- ☐ C Abba Alaba
- ☐ D Ankara

Question 4 ▲ Which of the following events are taking place during the year 1800?

- ☐ A Napoleon Bonaparte crosses the Alps with his army in Italy
- ☐ B First horse-drawn omnibuses are established in London
- ☐ C American Express is founded by Henry Wells & William Fargo
- ☐ D Aynag-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
---	---	---	---	---


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^AT_EX 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 ☐ Dakar
☐ Dili ☒ Dublin ☐ Džibouti

Question 4 What is the capital of Thailand?

☐ Beijing ☐ Bamjal ☒ 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 ☐ Dakar
☐ Dili ☐ Doha ☒ Dublin

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 \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 \LaTeX run updates the document but not the `.xy` file.

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

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

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

`francais` asks for french localisation.

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

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

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

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

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

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

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

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

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

3.2 Questions and answers

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

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

`question` Simple questions are enclosed in a `{question}\langle id \rangle` environment, and multiple questions are enclosed in a `{questionmult}\langle id \rangle` environment. These environments contain the question text, and the proposed choices inside a `choices`-like environment (see next). The `\langle id \rangle` 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^AT_EX 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^AT_EX

content.

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.

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 \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}
```

Question 1	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).

scorewrong= $\langle num \rangle$ gives the score for a wrong answer (defaults to 0).

keepas= $\langle name \rangle$ keeps the value entered by the student in variable $\{\langle name \rangle\}$, for future use with *alsocorrect* in another question.

alsocorrect= $\langle expression \rangle$ gives another acceptable answer, that can be based on the values entered by the student in the previous questions.

The text added at the end of the questions using `\AMCnumericChoices` when not in the separate answer sheet (and when a separate answer sheet is requested by the `separateanswersheet` package option) can also be set redefining the `\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 L^AT_EX package **tikz** must be loaded.

width for the width of the boxes.

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

`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.

`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}
```

Question 1 $2 + 2 =$

Ⓐ 1 Ⓑ 4 Ⓒ 10

```
\end{question}
```

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="display: inline-block; vertical-align: middle; text-align: center;"> <div style="border-left: 1px solid black; padding-left: 5px; margin-left: 5px;">Question 1</div> <div style="margin-left: 10px;">2+2=</div> <div style="display: inline-block; vertical-align: middle;"> $\left(\begin{array}{ccc} \square & 4 & \square \\ \square & 3 & \square \\ \square & 2 & \square \end{array} \right)$ </div> </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}
8 \RequirePackage{csvsimple}
9 \RequirePackage{environ}
10 % \end{macrocode}
11 %
12 % First, we read the options that can be given by AMC through the
13 % |jobname-config.tex| file:
14 % \begin{macrocode}
```

```

15 \InputIfFileExists{\jobname-config.tex}%
16 {\message{Loading configuration file...^^J}}{}

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

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

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

```

4.1 Variables

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

The package defines the following counters:

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

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

`\AMCid@etud` current student sheet number.

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

`\AMCid@check` current page checking number.

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

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

It also defines the following switches:

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

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

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

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

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

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

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

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

`\ifAMC@asqbloc` if questions are to be included in \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 \LaTeX boxes (so that they can't be splitted on two different columns for example).

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

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

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

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

`\ifAMC@calibration` if this \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.


```

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

32 \newif\ifAMC@ordre\AMC@ordrefalse
33 \newif\ifAMC@shuffle\AMC@shuffleGtrue
34 \newif\ifAMC@fullGroups\AMC@fullGroupsfalse
35 \newif\ifAMC@correthead\AMC@corretheadfalse
36 \newif\ifAMC@affichekeys\AMC@affichekeysfalse
37 \newif\ifAMC@keyline\AMC@keylinefalse
38 \newif\ifAMC@correc\AMC@correcfalse
39 \newif\ifAMC@textPos\AMC@textPosfalse
40 \newif\ifAMC@extractOnly\AMC@extractOnlyfalse
41 \newif\ifAMC@qbloc\AMC@qblocfalse
42 \newif\ifAMC@asqbloc\AMC@asqblocfalse
43 \newif\ifAMC@rbloc\AMC@rblocfalse
44 \newif\ifAMC@complete@multi\AMC@complete@multifalse
45 \newif\ifAMC@questionNumber\AMC@questionNumbertrue
46 \newif\ifAMC@calibration\AMC@calibrationfalse
47 \newif\ifAMC@catalog\AMC@catalogfalse
48 \newif\ifAMC@plain\AMC@plainfalse
49 \newif\ifAMC@bonne
50 \newif\ifAMC@type@multi
51 \newif\ifAMC@watermark\AMC@watermarktrue
52 \newif\ifAMC@inside@box\AMC@inside@boxfalse
53 \newif\ifAMC@outside@box\AMC@outside@boxfalse
54 \newif\ifAMC@ensemble\AMC@ensemblefalse
55 \newif\ifAMC@inside@digit\AMC@inside@digitfalse
56 \newif\ifAMC@formulaire@dedans\AMC@formulaire@dedansfalse
57 \newif\ifAMC@zoneformulaire
58 \newif\ifAMC@pagelayout\AMC@pagelayouttrue
59 \newif\ifAMC@postcorrect\AMC@postcorrectfalse
60 \newif\ifAMC@automarks\AMC@automarksfalse
61 \newif\ifAMC@invisible\AMC@invisiblefalse
62 \newif\ifAMC@pdfform\AMC@pdfformfalse
63 \let\AMCcompleteMulti=\AMCcomplete@multittrue
64 \let\AMCnoCompleteMulti=\AMCcomplete@multifalse

```

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

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

4.2 Dimensions

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

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

`\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.

```
66 \newdimen\AMCformVSpace\AMCformVSpace=1.2ex
67 \newdimen\AMCformHSpace\AMCformHSpace=.3em
68 \newdimen\AMCinterIrep\AMCinterIrep=\z@
69 \newdimen\AMCinterBrep\AMCinterBrep=.5ex
70 \newdimen\AMCinterIquest\AMCinterIquest=\z@
71 \newdimen\AMCinterBquest\AMCinterBquest=3ex
72 \newdimen\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).

```
73 \newif\ifAMCids@top
74 \newif\ifAMCids@side
75 \newdimen\AMCids@width
76 \newdimen\AMCids@height
77 \define@choicekey*{AMCids}{pos}[\AMCidsVar\AMCidsVarN]{none,top,side}{%
78   \ifcase\AMCidsVarN\relax
79     \AMCids@topfalse\AMCids@sidefalse
80   \or
81     \AMCids@toptrue\AMCids@sidefalse
82   \or
83     \AMCids@topfalse\AMCids@sidetrue
84   \fi
85 }
86 \define@key{AMCids}{width}{\AMCids@width=#1}
87 \define@key{AMCids}{height}{\AMCids@height=#1}
88 \def\AMCidsPosition#1{\setkeys{AMCids}{#1}}
89 \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}{\langle text \rangle}` sets the text to be printed behind each page of a draft exam.

```
90 \def\AMCtext#1#2{\expandafter\def\csname AMC@loc@#1\endcsname{#2}}
91 \def\AMClocalized#1{\csname AMC@loc@#1\endcsname}
```

4.4.1 English

Text indicating draft exams:

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

Message at page bottom when compiled out of AMC gui:

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

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

```
95 \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):

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

Headers for corrected version and catalog:

```
97 \def\AMC@loc@corrected{Corrected}
98 \def\AMC@loc@catalog{Catalog}
```

Localization text for Explanation

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

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

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

Word for 'question', singular and plural forms:

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

Default text to write in the students' name box:

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

4.4.2 Catalan

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

```
104 \def\AMC@loc@CA{
105   \def\AMC@loc@draft{PROJECTE}
106   \def\AMC@loc@message{Pel vostre examen, imprimiu preferiblement
107     els documents compilats amb l'ajuda de auto-multiple-choice.}
108   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
109   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
110   \def\AMC@loc@corrected{Correcci\`o}
```

```

111 \def\AMC@loc@catalog{Cat\‘aleg}
112 \def\AMC@loc@explain{\textit{\textbf{Explicaci\‘o : }}}
113 \def\AMC@loc@none{Cap de les respostes \‘es correcte.}
114 \def\AMC@loc@question{pregunta}
115 \def\AMC@loc@questions{preguntes}
116 \def\AMC@loc@namesurname{Nom i cognoms:}
117 }

```

4.4.3 Dutch

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

```

118 \def\AMC@loc@NL{
119 \def\AMC@loc@draft{Ontwerp}
120 \def\AMC@loc@message{Gebruik bij uw proefwerk bij voorkeur die
121 documenten welke door auto-multiple-choice zijn aangemaakt.}
122 \def\AMC@loc@qf##1{\textbf{Vraag ##1 :}}
123 \def\AMC@loc@q##1##2{\textbf{Vraag ##1} ##2}
124 \def\AMC@loc@corrected{Correctie}
125 \def\AMC@loc@catalog{Catalogus}
126 \def\AMC@loc@none{Geen van de antwoorden is juist.}
127 \def\AMC@loc@question{vraag}
128 \def\AMC@loc@questions{vragen}
129 \def\AMC@loc@namesurname{Achternaam en voornaam:}
130 }

```

4.4.4 French

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

```

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

```

4.4.5 German

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

```

145 \def\AMC@loc@DE{
146 \def\AMC@loc@draft{ENTWURF}
147 \def\AMC@loc@message{Benutzen Sie f\“ur Ihre Pr\“ufung bevorzugt Dokumente die mit

```

```

148   auto-multiple-choice erstellt wurden.}
149 \def\AMC@loc@qf##1{\textbf{Frage ##1 :}}
150 \def\AMC@loc@q##1##2{\textbf{Frage ##1} ##2}
151 \def\AMC@loc@corrected{Korrektur}
152 \def\AMC@loc@catalog{Katalog}
153 \def\AMC@loc@explain{\textit{\textbf{Erkl\"arung : }}}
154 \def\AMC@loc@none{Keine dieser Antworten ist korrekt.}
155 \def\AMC@loc@question{Frage}
156 \def\AMC@loc@questions{Fragen}
157 \def\AMC@loc@namesurname{Vor- und Nachname:}
158 }

```

4.4.6 Italian

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

```

159 \def\AMC@loc@IT{
160   \def\AMC@loc@draft{BOZZA}
161   \def\AMC@loc@message{Per l'esame, \textit{e} preferibile stampare i documenti
162     a partire da auto-multiple-choice.}
163   \def\AMC@loc@qf##1{\textbf{Domanda ##1:}}
164   \def\AMC@loc@q##1##2{\textbf{Domanda ##1} ##2}
165   \def\AMC@loc@corrected{Correzione}
166   \def\AMC@loc@catalog{Catalogo}
167   \def\AMC@loc@none{Nessuna risposta \textit{e} giusta.}
168   \def\AMC@loc@question{domanda}
169   \def\AMC@loc@questions{domande}
170   \def\AMC@loc@namesurname{Nome e cognome:}
171 }

```

4.4.7 Norwegian

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

```

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

```

4.4.8 Portuguese

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

```

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

```

4.4.9 Spanish

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

```

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

```

4.4.10 Japanese

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

```

211 \def\AMC@loc@JA{
212   \def\AMC@loc@draft{◇◇◇◇}
213   \def\AMC@loc@message{◇◇◇◇◇◇◇◇auto-multiple-choice◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇◇}
214   \def\AMC@loc@qf##1{\textbf{◇##1:}}
215   \def\AMC@loc@q##1##2{\textbf{◇##1} ##2}
216   \def\AMC@loc@corrected{◇◇◇◇}
217   \def\AMC@loc@catalog{◇◇◇◇◇◇}
218   \def\AMC@loc@explain{\textit{\textbf{◇◇: }}}
219   \def\AMC@loc@none{◇◇◇◇◇◇}
220   \def\AMC@loc@question{◇}
221   \def\AMC@loc@questions{◇}
222 }

```

4.4.11 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:

```
223 \AtBeginDocument{\@ifpackageloaded{cleveref}{%
224   \message{AMC/cleveref integration loaded^^J}%
225   \creffalias{AMCquestionaff}{question}%
226   \crefname{question}{\AMC@loc@question}{\AMC@loc@questions}%
227 }{}}%
```

4.6 Random

4.6.1 Random pseudo-generator

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

```
228 \ifx\AMC@SR\undefined\newcount\AMC@SR\fi
229 \providecommand\AMC@SRconst{2097152}
230 \providecommand\AMC@SRset[1]{\global\AMC@SR#1 \ignorespaces}
231 \providecommand\AMC@SRadvance{%
232   \begingroup%
233   \ifnum\AMC@SR<\AMC@SRconst\relax\AMC@SR@count\z@%
234   \ifodd\AMC@SR\advance\AMC@SR@count\@ne\fi%
235   \global\divide\AMC@SR\tw@%
236   \ifodd\AMC@SR@count\global\advance\AMC@SR\AMC@SRconst\relax\fi%
237   \endgroup}
238 \providecommand\AMC@SRbit{\AMC@SRadvance\ifodd\AMC@SR1\else0\fi}
239 \providecommand\AMC@SRtest[2]{\AMC@SRadvance%
240   \ifodd\AMC@SR#2\else#1\fi\ignorespaces}
241 \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>}`.

```
242 \AMC@SRset{1515}
243 \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.

```
244 \newcount\AMC@SR@count
245 \def\AMC@SR@time{\AMC@SRset{\time}}
246 \newcount\AMC@SRnum
247 \def\AMC@SRnextByte{\AMC@SRnum=\z@%
248   \AMC@SR@count=20%
```

```

249 \loop\multiply\AMC@SRnum\tw%
250 \AMC@SRtest{\advance\AMC@SRnum\@ne}{}%
251 \ifnum\AMC@SR@count>\@ne\advance\AMC@SR@count\m@ne\repeat%
252 }
253 \newcommand\AMC@SRmax[1]{\AMC@SRnextByte%
254 \AMC@SR@count=\AMC@SRnum%
255 \divide\AMC@SR@count by #1\relax%
256 \multiply\AMC@SR@count by #1\relax%
257 \advance\AMC@SRnum by -\AMC@SR@count%
258 }

```

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).

```

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

```

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).

```

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

```



```

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

```

4.8 Boxes

4.8.1 Character logging

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

```

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

```

4.8.2 Position logging

`\AMC@tracebox` Command `\AMC@tracebox{<trace>}{<key>}{<content>}` makes a L^AT_EX 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.

```

292 \def\AMC@shapename@{\ifAMC@invisible none\else\AMC@shapename\fi}
293 \def\AMC@tracepos#1#2{%
294 \ifAMC@calibration\ifx\@empty#1\@empty\else%
295 \pdfsavepos\protected@write\AMC@XYFILE{}{%
296 \string\tracepos%
297 {\the\AMCid@etud/\thepage:#2}%
298 {\noexpand\number\pdflastxpos sp}%
299 {\noexpand\number\pdflastypos sp}%
300 {\AMC@shapename}}%
301 \fi\fi}
302 \def\AMC@traceposx#1#2{%
303 \ifAMC@calibration\ifx\@empty#1\@empty\else%
304 \pdfsavepos\protected@write\AMC@XYFILE{}{%
305 \string\tracepos%
306 {\the\AMCid@etud/\thepage:#2}%
307 {\noexpand\number\pdflastxpos sp}%
308 {0sp}%
309 {\AMC@shapename}}%
310 \fi\fi}
311 \def\AMC@traceposy#1#2{%
312 \ifAMC@calibration\ifx\@empty#1\@empty\else%
313 \pdfsavepos\protected@write\AMC@XYFILE{}{%

```

```

314 \string\tracepos%
315 {\the\AMCid@etud/\thepage:#2}%
316 {0sp}%
317 {\noexpand\number\pdfloatypos sp}%
318 {\AMC@shapename}}}%
319 \fi\fi}
320 \newcommand\AMC@tracebox[3]{%
321 \vbox{\AMC@traceposy{#1}{#2}%
322 \hbox{\AMC@traceposx{#1}{#2}#3\AMC@traceposx{#1}{#2}}}%
323 \AMC@traceposy{#1}{#2}}}%
324 \def\AMC@pagepos{%
325 \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
326 \string\page%
327 {\the\AMCid@etud/\thepage/\the\AMCid@check}%
328 {\the\paperwidth}{\the\paperheight}}\fi}

```

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

```

329 \newcommand{\AMCdontScan}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontscan{\the\AMCid@etud,\th
330 \newcommand{\AMCdontAnnotate}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontannotate{\the\AMCid@
331 %

```

`\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.

```

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

```

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

```

339 \newwrite\AMC@XYspecial
340 \newwrite\AMC@tmpXY
341 \newenvironment{amcxyfile}[1]{%
342 \openout\AMC@XYspecial#1%
343 \let\AMC@tmpXY=\AMC@XYFILE%
344 \let\AMC@XYFILE=\AMC@XYspecial%
345 }{\let\AMC@XYFILE=\AMC@tmpXY\closeout\AMC@XYspecial}

```

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

```

346 \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 to retrieve it easily from the scans.

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

```

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

As an example,

```

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

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

```

produces the following box:

Name:

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

4.8.3 Boxes to be checked by students

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


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

```

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

```

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

```

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

```

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

```

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

```

```

396 \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
397 \fcolorbox{\AMC@boxcolor@}{\AMC@fillcolor@}%
398 {%
399   \boxput*(0,0){%
400     \ifAMC@draw@cross\AMC@crosschar\fi%
401   }{%
402     \vbox to \AMC@boxedheight{%
403       \AMC@tracepos{#3}{#4}%
404       \vfill%
405       \hbox to \AMC@boxedwidth{\hfill%
406         \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
407         \AMC@smashcentered{#2}%
408         \hfill}\vfill}}%
409     \AMC@tracepos{#3}{#4}}%
410 }

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

411 \newcommand\AMC@makeovalbox[3]{%
412   \AMC@setcolors@{#1}{#2}%
413   \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
414   \AMC@save@box{#3}{%
415     \begin{tikzpicture}%
416       \useasboundingbox (-0.5\AMC@boxedwidth-0.5\AMC@boxedrule,0.5\AMC@boxedheight+0.5\AMC@boxedrule)
417       rectangle (0.5\AMC@boxedwidth+0.5\AMC@boxedrule,-0.5\AMC@boxedheight-0.5\AMC@boxedrule);
418       \draw[\AMC@boxcolor@,fill=\AMC@fillcolor@,line width=\AMC@boxedrule,rounded corners=\AMC@oval@radius]
419       (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight)
420       rectangle (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
421       \ifAMC@draw@cross
422         \draw[\AMC@boxcolor@,line width=\AMC@crossrule]
423         (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight)
424         (0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (-0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
425       \fi
426     \end{tikzpicture}}%
427 }

428 \newcommand\AMC@shapeprepare@oval{%
429   \ifx\AMC@ovalbox@R\undefined\else%
430     \AMC@makeovalbox{1}{\AMC@ovalbox@R}%
431     \AMC@makeovalbox{1}{\AMC@checkedbox}{\AMC@ovalbox@RF}%
432     \AMC@makeovalbox{}{\AMC@ovalbox@}%
433     \AMC@makeovalbox{}{\AMC@checkedbox}{\AMC@ovalbox@F}%
434   \fi%
435 }

436 \newcommand\AMC@shape@oval[4]{%
437   \AMC@setcolors@{#3}{#2}%
438   \AMC@tracebox{#3}{#4}{\boxput*(0,0){%
439     \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
440     \AMC@smashcentered{#2}%
441   }{%
442     \ifx\@empty#3\@empty%
443       \ifx #2\AMC@checkedbox%
444         \AMC@use@box{\AMC@ovalbox@F}%

```

```

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

```

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

```

```
486 \else #1\fi}{#2}{#3}{#4}}
```

`\AMCboxStyle` The dimensions of these box are managed by `\AMCboxDimensions{<sizes>}`, where `<sizes>` is a comma separated list of `<name>=<dimension>` constructs. Here, `<name>` 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 `<color>` 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{<color>}` command is defined as an alias to `\AMCboxStyle{color=<color>}`, and `\AMCboxDimensions` as an alias to `\AMCboxStyle`, for backward compatibility.

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

<code>\AMCboxOutsideLetter</code>	Command <code>\AMC@box{<char>}{<answer>}</code> prints a box with character <code><char></code> inside, showing answer
<code>\AMC@box</code>	<code><answer></code> (<code>\AMC@checkbox</code> to get a filled box), using global variables to identify the box (question
<code>\AMC@formBox@</code>	and choice).
<code>\AMC@formBox</code>	It calls <code>\AMC@formBox@{<char>}{<answer>}{<trace>}{<key>}</code> to actually render the box.
<code>outsideLabelFormat</code>	Command <code>\AMC@formBox</code> simply sets the first argument when empty before calling <code>\AMC@formBox@</code> .

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`).

```

518 \def\AMCoutsideLabelFormat#1{#1}
519 \newcommand\AMCboxOutsideLetter[2]{#1\nobreak\hspace{.1em}\AMCoutsideLabelFormat{#2}}
520 \newif\ifAMC@printformoutside%
521 \newcommand\ifAMC@printformoutside{%
522   \AMC@printformoutside@false%
523   \ifAMC@ensemble\ifAMC@outside@box%
524     \ifAMC@formulaire@dedans\AMC@printformoutside@true\fi%
525     \ifAMC@zoneformulaire\AMC@printformoutside@true\fi%
526   \fi\fi%
527   \ifAMC@printformoutside%
528 }
529 \newcommand\AMC@formBox@[4]{%
530   \ifAMC@printformoutside% letter to be written outside the box
531     \AMCboxOutsideLetter{\AMC@answerBox@}{#2}{#3}{#4}{#1}%
532   \else%
533     \AMC@answerBox@{#1}{#2}{#3}{#4}%
534   \fi%
535   \AMC@tracechar{#1}{#2}{#3}{#4}%
536 }
537 \newif\ifAMC@printkeyoutside%
538 \newcommand\ifAMC@printkeyoutside{%
539   \AMC@printkeyoutside@false%
540   \ifAMC@ensemble%
541     \ifAMC@outside@box\AMC@printkeyoutside@true\fi%
542   \else%
543     \ifAMC@inside@box\else\AMC@printkeyoutside@true\fi%
544   \fi%
545   \ifAMC@printkeyoutside%
546 }
547 \newcommand\AMC@keyBox@[4]{%
548   \ifAMC@printkeyoutside%
549     \AMCboxOutsideLetter{\AMC@answerBox@}{#2}{#3}{#4}{#1}%
550   \else%
551     \AMC@answerBox@{#1}{#2}{#3}{#4}%
552   \fi%
553   \AMC@tracechar{#1}{#2}{#3}{#4}%
554 }
555 \newcommand\AMC@formBox[4]{%
556   \AMC@formBox@{\ifx\@empty#1\@empty%
557     \AMCchoiceLabel{AMC@ncase}%
558     \else #1\fi}{#2}{#3}{#4}%
559 }
560 \newcommand{\AMC@box}[2]{%
561   \ifAMC@ensemble%
562     \ifAMC@zoneformulaire% for codes inside form sheet

```



```

563     \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
564 \else%
565     \ifAMCformulaire@dedans% for answer boxes inside form sheet
566     \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
567     \else% outside form sheet: not to be read during data capture
568     \AMC@formBox{#1}{#2}{1}{casequestion:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
569 \fi\fi%
570 \else% no separate sheet for answers: always read
571 \ifAMC@inside@box%
572     \AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
573 \else%
574     \AMC@formBox@{}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
575 \fi%
576 \fi%
577 }

```

4.8.4 Scoring zones

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

```

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

```

4.8.5 Binary boxes

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

`\AMCid@checkmax` The check number is just decreased each page. Its maximum value is `\AMCid@checkmax`. 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`.

```

594 \def\AMCid@checkmax{60}
595 \def\AMC@NCBetud{12}
596 \def\AMC@NCBpage{6}

```

```

597 \def\AMC@NCBcheck{6}
598 \newlength{\AMC@CBtaille}\setlength{\AMC@CBtaille}{5cm}
599 \def\AMC@premierecopie{1}

```

\AMC@binaryBoxes Command `\AMC@binaryBoxes[$\langle ndigits \rangle$]{ $\langle n \rangle$ }` prints $\langle ndigits \rangle$ boxes to represent number $\langle n \rangle$ 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:



```

600 \newtoks\AMCbin@sequence
601 \newcount\AMCbin@number
602 \newcount\AMCbin@ndigits
603 \newcount\AMCbin@id
604 \newcount\AMCbin@digit
605 \def\AMCbin@one{\advance\AMCbin@digit\@ne%
606   \AMC@answerBox@{\AMC@checkbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@digit}}
607 \def\AMCbin@zero{\advance\AMCbin@digit\@ne%
608   \AMC@answerBox@{\AMC@checkbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@digit}}
609 \def\AMCbin@begin#1{\AMCbin@id=#1\AMCbin@digit=\z@%
610 \newcommand{\AMC@binaryBoxes}[2][1]{%
611 {\AMCboxDimensions{shape=square,size=.32cm,down=0pt,rule=.2pt,cross=false}\AMCbin@sequence={}\AMCbin@number=
612 \AMCbin@ndigits=\z@%
613 \loop%
614 \ifnum\AMCbin@number>\z@%
615 \advance\AMCbin@ndigits\@ne%
616 \ifodd\AMCbin@number\AMCbin@sequence=\expandafter{\expandafter\AMCbin@one\the\AMCbin@sequence}%
617 \else\AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\fi%
618 \divide\AMCbin@number\tw@%
619 \repeat%
620 \loop\relax%
621 \ifnum\AMCbin@ndigits<#1\advance\AMCbin@ndigits\@ne%
622 \AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\repeat%
623 \the\AMCbin@sequence%
624 \ifnum\AMCbin@ndigits>#1\PackageError{automultiplechoice}{Too low AMC@NCB value (got #1 but needs \the\AMCbin
625 }}

```

4.9 Checking Environment

\AMCcurrentenv Sets the current environment as document.

```

626 \def\AMCcurrentenv{document}

```

\AMCif@env Checks for the current environment.

```

627 \def\AMCif@env#1{
628   \def\AMC@tempenv{#1}%
629   \ifx\AMC@tempenv\AMCcurrentenv
630     \expandafter\@firstoftwo
631   \else
632     \expandafter\@secondoftwo
633   \fi
634 }

```

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.

`\nouveau groupe` Command `\nouveau groupe{<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, or two successive `questions` to be kept together, or anything else. Calling command `\nouveau groupe` is not compulsory, as `\element` calls it if necessary.

```

635 \newcount\AMCtok@k
636 \newcount\AMCtok@max
637 \newcount\AMCtok@size
638 \newcommand{\nouveau groupe}[2]{%
639   \expandafter\ifx\csname #1@k\endcsname\relax%
640     \expandafter\newcount\csname #1@k\endcsname%
641     \expandafter\newcount\csname AMC#1@j\endcsname%
642     \csname #1@k\endcsname=\z@ \relax%
643     \csname AMC#1@j\endcsname=\z@ \relax%
644     \setgroupmode{#1}{\AMCdefault@groupmode}%
645   \fi%
646 }
647 \newcommand\AMC@prepare@element[1]{%
648   \nouveau groupe{#1}{}%
649   \global\advance\csname #1@k\endcsname \@ne \relax%
650   \AMCtok@k=\csname #1@k\endcsname%
651   \expandafter\ifx\csname #1@\romannumeral\AMCtok@k\endcsname\relax%
652     \expandafter\newtoks\csname #1@\romannumeral\AMCtok@k\endcsname\fi%
653 }
654 \newcommand{\element}[2]{%
655   \AMC@prepare@element{#1}%
656   \global\csname #1@\romannumeral\AMCtok@k\endcsname={#2}%
657 }

```

`\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.

```

658 \def\AMCdefault@groupmode{fixed}
659 \newcommand{\setdefaultgroupmode}[1]{\def\AMCdefault@groupmode{#1}}
660 \newcommand{\setgroupmode}[2]{%
661   \expandafter\ifx\csname AMCgrouppre@#2\endcsname\relax%
662     \PackageError{automultiplechoice}{Unknown group mode for #1 : #2}%
663     {You asked to set group '#1' mode to '#2',
664      but '#2' is not a valid group mode}%
665   \else%
666     \expandafter\global\expandafter\def\csname AMC#1@mode\endcsname{#2}%
667   \fi%
668 }

```

The functions `\AMCgrouppre@xxx{<group-name>}{<n>}{<i>}` are called before using $\langle n \rangle$ elements from group $\langle group-name \rangle$ starting from index $\langle i \rangle$ (negative value for $\langle i \rangle$ 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).

```

669 \newcommand{\AMCgrouppre@fixed}[3]{%
670   \ifnum#3<\z@%
671     \csname AMC#1@j\endcsname=\z@%
672   \else%
673     \csname AMC#1@j\endcsname=#3%
674   \fi%
675 }

```

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.

```

676 \newcommand{\AMCgrouppre@withreplacement}[3]{%
677   \ifnum#3<\z@%
678     \csname AMC#1@j\endcsname=\z@%
679   \else%
680     \csname AMC#1@j\endcsname=#3%
681   \fi%
682   \shufflegroup{#1}%
683 }

```

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.

```

684 \newcount\AMC@imax
685 \newcommand{\AMCgrouppre@withoutreplacement}[3]{%
686   \ifnum#3<\z@%
687   \else%
688     \csname AMC#1@j\endcsname=#3%
689   \fi%
690   \ifnum\AMCtok@ik=\AMCloop@k%
691     \AMCtok@ik=\z@%
692   \fi%
693   \ifnum\AMCtok@ik=\z@%
694     \shufflegroup{#1}%
695   \else%

```

```

696 \AMC@imax=\AMCloop@k%
697 \advance\AMC@imax -#2\relax%
698 \ifnum\AMCtok@ik>\AMC@imax%
699 \shufflegroupslice{#1}{\@ne}{\AMCtok@ik}%
700 \ifnum\AMCtok@ik<\AMCloop@k%
701 \advance\AMCtok@ik\@ne%
702 \shufflegroupslice{#1}{\AMCtok@ik}{\AMCloop@k}%
703 \fi%
704 \fi%
705 \fi%
706 }

```

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

```

707 \newcommand{\AMCgrouppre@cyclic}[3]{%
708 \ifnum#3<z@%
709 \else%
710 \csname AMC#1@j\endcsname=#3%
711 \fi%
712 }

```

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

```

713 \newcommand{\AMCgroup@pre}[4]{%
714 \csname AMCgrouppre@#1\endcsname{#2}{#3}{#4}%
715 }

```

`\shufflegroup` Command `\shufflegroup{<group-name>}` shuffles the elements of group `<group-name>`, and
`\insertgroup` `\shufflegroupslice{<group-name>}{<a>}{}` shuffles elements `<a>` to `` 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.

```

716 \newcommand{\shufflegroup}[1]{%
717 \ifAMC@shuffleG{\AMC@shuffletoks{\number\csname #1@k\endcsname}{#1@}}\fi%
718 }
719 \newcommand{\shufflegroupslice}[3]{%
720 \ifAMC@shuffleG{\AMC@shuffletoks[#2]{#3}{#1@}}\fi%
721 }
722 \newcount\AMCtok@ik
723 \newcount\AMCloop@k
724 \newcommand{\AMCgrouploop@prep}[3]{%
725 \AMCtok@size=#1\relax%
726 \ifAMC@fullGroups\AMCtok@size=z@\fi%
727 \ifnum\AMCtok@size<\@ne%
728 \AMCtok@size=\csname #2@k\endcsname%
729 \fi%
730 \AMCtok@ik=\csname AMC#2@j\endcsname%
731 \AMCloop@k=\csname #2@k\endcsname%
732 \expandafter\ifx\csname AMC#2@mode\endcsname\relax%

```

```

733 \PackageError{automultiplechoice}{No group mode for #2}%
734 {No mode has been defined for group '#2'. This should not occur...}%
735 \fi%
736 \AMCgroup@pre{\csname AMC#2@mode\endcsname}{#2}{\the\AMCtok@size}{#3}%
737 }
738 \newcommand{\AMCgrouploop@next}[1]{%
739 \global\advance\csname AMC#1@j\endcsname\@ne\relax%
740 \expandafter\ifnum\csname AMC#1@j\endcsname>\AMCloop@k\relax%
741 \global\csname AMC#1@j\endcsname=\@ne%
742 \fi%
743 \AMCtok@ik=\csname AMC#1@j\endcsname%
744 \advance\AMCtok@size\m@ne%
745 }
746 \newcommand{\insertgroupfrom}[3][0]{%
747 \AMCgrouploop@prep{#1}{#2}{#3}%
748 {\loop%
749 \AMCgrouploop@next{#2}%
750 {\the\csname #2@romannumeral\AMCtok@ik\endcsname}%
751 \ifnum\AMCtok@size>\z@repeat}%
752 }
753 \newcommand{\insertgroup}[2][0]{%
754 \insertgroupfrom[1]{#2}{-1}%
755 }

```

`\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>`, erasing 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>}` 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.

```

756 \newcommand{\cleargroup}[1]{%
757 \nouveau groupe{#1}{}%
758 \csname #1@k\endcsname=\z@relax%
759 \csname AMC#1@j\endcsname=\z@relax%
760 }
761 \newcommand{\copygroupfrom}[4][0]{%
762 \AMCgrouploop@prep{#1}{#2}{#4}%
763 {\loop%
764 \AMCgrouploop@next{#2}%
765 \AMC@prepare@element{#3}%
766 \global\csname #3@romannumeral\AMCtok@k\endcsname=\csname #2@romannumeral\AMCtok@ik\endcsname%
767 \ifnum\AMCtok@size>\z@repeat}%
768 }
769 \newcommand{\copygroup}[3][0]{%
770 \copygroupfrom[1]{#2}{#3}{-1}%

```

771 }

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`.

```
772 \newcount\AMCrep@count
773 \AMCload@counter=199
774 \@whilenum\AMCload@counter>0\do{%
775   \expandafter\newtoks\csname reponse@\romannumeral\AMCload@counter\endcsname%
776   \advance\AMCload@counter\m@ne%
777 }
```

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

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

```
778 \newcommand\AMCload@reponse[2]{%
779   \global\advance\AMCload@counter\@ne\relax%
780   \global\csname reponse@\romannumeral\AMCload@counter\endcsname%
781   =\expandafter{\expandafter\AMCrep@count\expandafter=#2 #1}%
782 }
783 \newcommand\AMCrien@deux[2]{#1}
```

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

```
784 \def\shuffle@it{\AMC@shuffletoks{\number\AMCload@counter}{reponse@}}
785 \newcount\AMCnum@questions
786 \newcommand\AMCdump@reponses{%
787   \global\AMCnum@questions=\AMCload@counter%
788   \@whilenum\AMCload@counter>0\do{%
789     \the\csname reponse@\romannumeral\AMCload@counter\endcsname%
790     \advance\AMCload@counter\m@ne}}
```

4.11.1 Managing answers

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

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

- If $\langle mode \rangle = o$, `\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`.

```

791 \newcommand\AMCrep@init[1]{%
792   \ifAMC@ordre\AMCrep@o\else%
793     \csname AMCrep@#1\endcsname\fi\AMCload@counter=\z@}
794 \newcommand\AMCrep@o{%
795   \def\AMCload@reponse{\AMCrien@deux}\def\AMCrep@fini{}}
796 \newcommand\AMCrep@r{%
797   \def\AMCload@reponse{\AMCload@reponse}%
798   \def\AMCrep@fini{\shuffle\it\AMCdump@reponses}}
799 \newcount\AMCrep@count
800 \newcommand\lastchoices{%
801   \AMCrep@count=\AMCrep@count%
802   \AMCrep@fini\AMCrep@init{o}%
803   \AMCrep@count=\AMCrep@count}
804 \newcommand\@aucune{\emph{\AMC@loc@none}}
805 \newcommand\AMC@fin@rep{%
806   \ifAMCcomplete@multi\ifAMCtype@multi%
807     \lastchoices\AMCrep@count=-1%
808     \ifAMC@bonne\wrongchoice{\@aucune}\else%
809       \ifAMC@postcorrect\wrongchoice{\@aucune}\else\correctchoice{\@aucune}\fi%
810     \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{\langle number \rangle}` is responsible for announcing question, and `\AMCformAnswer{\langle box \rangle}` is responsible for printing the box to be ticked, given as argument $\langle box \rangle$.

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

```

811 \def\AMCformBeforeQuestion{\vspace{\AMCformVSpace}\par}
812 \def\AMCformAfterQuestion{\ifAMC@asqbloc\egroup\fi}
813 \def\AMCformQuestion#1{\AMC@loc@qf{#1}}
814 \def\AMCformQuestionN{\AMCformQuestion{\AMC@qaff}}
815 \def\AMCformQuestionA{%
816   \setcounter{AMC@ncase}{0}%
817   \AMCformBeforeQuestion%

```



```

818 \ifAMC@asqbloc\vbox\bgroup\fi%
819 \ifx\@empty\AMC@sza@callout\@empty\else%
820 \csname\AMC@sza@callout\endcsname%
821 \fi%
822 \AMCformQuestionN%
823 \ifx\@empty\AMC@sza@callin\@empty\else%
824 \csname\AMC@sza@callin\endcsname%
825 \fi%
826 }
827 \def\AMCformAnswer#1{\hspace{\AMCformHSpace} #1}
828 \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 `\AMCform` `\AMC@mem@openQuestion` adds to memory question code to announce current question.

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

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

```

829 \ExplSyntaxOn
830
831 \prg_set_conditional:Nnn \amc_if_separate_question: { p , T } {
832   \ifAMC@ensemble
833     \ifAMC@zoneformulaire
834       \prg_return_false:
835     \else
836       \prg_return_true:
837     \fi
838   \else
839     \prg_return_false:
840   \fi
841 }
842 \cs_new_eq:NN \AMC@if@separate@question \amc_if_separate_question:T
843
844 \int_new:N \amc_memory_elts_count
845
846 \cs_new:Nn \amc_clear_memory: { \int_gzero:N \amc_memory_elts_count }
847 \cs_new_eq:NN \AMC@mem@clear \amc_clear_memory:
848
849 \cs_new:Npn \amc_memory_elt_i:n #1 {
850   amc_memory_elts_ \int_to_alph:n { #1 }
851 }
852 \cs_new:Nn \amc_memory_current_elt: {
853   \amc_memory_elt_i:n \amc_memory_elts_count
854 }
855 \cs_new:Npn \amc_memory_vars_i:n #1 {
856   amc_memory_vars_ \int_to_alph:n { #1 }

```

```

857 }
858 \cs_new:Nn \amc_memory_current_vars: {
859   \amc_memory_vars_i:n \amc_memory_elts_count
860 }
861
862 \cs_new:Nn \amc_add_memory_elt: {
863   \int_gincr:N \amc_memory_elts_count
864   \tl_gclear_new:c { \amc_memory_current_elt: }
865   \tl_gclear_new:c { \amc_memory_current_vars: }
866 }
867 \cs_new_eq:NN \AMC@mem@next \amc_add_memory_elt:
868
869 \cs_new:Npn \amc_add_to_memory:n #1 {
870   \tl_gput_right:cn { \amc_memory_current_elt: } { #1 }
871 }
872 \cs_new_eq:NN \AMC@mem@add \amc_add_to_memory:n
873
874 \cs_new:Npn \amc_add_to_vars:n #1 {
875   \tl_gput_right:cn { \amc_memory_current_vars: } { #1 }
876 }
877 \cs_new_eq:NN \AMC@mem@addvar \amc_add_to_vars:n
878
879 \cs_new:Npn \amc_add_qidaffname:nnn #1#2#3 {
880   \amc_add_to_vars:n {\AMCid@quest=#1\setcounter{AMCquestionaff}{#2}%
881     \global\def\AMCid@name{#3}}
882 }
883 \cs_generate_variant:Nn \amc_add_qidaffname:nnn { xxx }
884 \cs_new_eq:NN \AMC@mem@qidaffname \amc_add_qidaffname:xxx
885
886 \cs_new:Npn \amc_mem_elt_cat:n #1 {
887   \amc_add_to_vars:n { \def\AMCmem@elt@cat{ #1 } }
888 }
889 \cs_generate_variant:Nn \amc_mem_elt_cat:n { x }
890 \cs_new_eq:NN \AMC@mem@category \amc_mem_elt_cat:x
891
892 \cs_new:Npn \amc_add_aid:n #1 {
893   \amc_add_to_memory:n {\AMCrep@count=#1}
894 }
895 \cs_generate_variant:Nn \amc_add_aid:n { x }
896 \cs_new_eq:NN \AMC@mem@aid \amc_add_aid:x
897
898 \cs_new:Npn \amc_if_category_is_p:n #1 {
899   \str_if_eq_p:on { \AMCmem@elt@cat } { #1 }
900 }
901 \cs_new:Npn \amc_use_memory:n #1 {
902   \int_step_inline:nnnn { 1 } { 1 } \amc_memory_elts_count {
903     \def\AMCmem@elt@cat{ plain }
904     \tl_use:c { \amc_memory_vars_i:n { ##1 } }
905     \bool_if:nTF { #1 } {
906       \tl_use:c { \amc_memory_elt_i:n { ##1 } }

```

```

907     } { }
908   }
909 }
910 \cs_new:Nn \amc_use_memory: { \amc_use_memory:n { \c_true_bool } }
911 \cs_new_eq:NN \AMC@mem@show \amc_use_memory:
912 \cs_new_eq:NN \AMC@mem@show@filter \amc_use_memory:n
913 \cs_new_eq:NN \AMC@category \amc_if_category_is_p:n
914
915 \ExplSyntaxOff
916 \newcommand\AMC@mem@add@ifneeded[1]{%
917   \AMC@if@separate@question{%
918     \AMC@mem@add{#1}%
919   }%
920 }
921 \newcommand\AMC@mem@addsingle@ifneeded[2]{%
922   \AMC@if@separate@question{%
923     \AMC@mem@next%
924     \AMC@mem@category{#2}%
925     \AMC@mem@add{#1}%
926   }%
927 }
928 \newcommand\AMC@mem@answer[1]{%
929   \addtocounter{AMC@ncase}{1}%
930   \AMC@if@separate@question{%
931     \AMC@mem@aid{\the\AMC@rep@count}%
932     \AMC@mem@add{\AMC@form@answerA{#1}}%
933   }%
934 }
935 \newcommand\AMC@mem@openQuestion{%
936   \AMC@if@separate@question{%
937     \AMC@mem@next%
938     \AMC@mem@qid@affname{\the\AMC@id@quest}{\arabic{AMC@question@aff}}{\AMC@id@name}%
939     \AMC@mem@add{\AMC@form@questionA}%
940   }%
941 }
942 \def\AMC@form@begin{%
943   \AMC@zone@formulairetrue\setcounter{section}{0}%
944   \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMC@pageFull}\fi\fi%
945 }
946 \newcommand\AMC@form{%
947   \ifAMC@ensemble\AMC@formulaire@dedanstrue%
948     \AMC@mem@show%
949   \fi}
950 \newcommand\AMC@form@filter[1]{%
951   \ifAMC@ensemble\AMC@formulaire@dedanstrue%
952     \AMC@mem@show@filter{#1}%
953   \fi}
954 \newif\ifAMC@keepmemory
955 \newcommand\AMC@formS{%
956   \ifAMC@ensemble\AMC@formulaire@dedanstrue%

```

```

957 \AMC@amclog{AUTOQCM[BR=0]^^J}\AMC@mem@show%
958 \AMC@keepmemorytrue%
959 \fi}

```

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

```

960 \newcommand{\AMCsectionNumbered}[1]{%
961 \section{#1}\AMC@mem@addsingle@ifneeded{\section{#1}}{section}}
962 \newcommand{\AMCsubsectionNumbered}[1]{%
963 \subsection{#1}\AMC@mem@addsingle@ifneeded{\subsection{#1}}{subsection}}
964 \newcommand{\AMCsectionStar}[1]{%
965 \section*{#1}\AMC@mem@addsingle@ifneeded{\section*{#1}}{section}}
966 \newcommand{\AMCsubsectionStar}[1]{%
967 \subsection*{#1}\AMC@mem@addsingle@ifneeded{\subsection*{#1}}{subsection}}
968 \def\AMCsection{\@ifstar\AMCsectionStar\AMCsectionNumbered}
969 \def\AMCsubsection{\@ifstar\AMCsubsectionStar\AMCsubsectionNumbered}

```

4.11.3 Formatting answers

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

`choicescustom` Use `\AMCBoxedAnswers` to request all answers to be included in L^AT_EX boxes; this can be useful for example when using multicolumn answers formatting.

```

tikz-single
tikz-multi
\AMCBoxedAnswers 970 \def\AMCBoxedAnswers{\AMC@rbloctrue}
971 \newenvironment{choices}[1][r]{%
972 \AMCrep@count=\z@\def\une@rep{\AMCrep@itemize}%
973 \ifAMC@rbloc\def\une@rep{\AMCrep@bloc}%
974 \else\begin{itemize}\setlength{\itemsep}{\AMCinterIrep}\fi%
975 \AMCrep@init{#1}}%
976 {\AMC@fin@rep\ifAMC@rbloc\else\end{itemize}\fi}
977 \newenvironment{choiceshoriz}[1][r]{%
978 \AMCrep@count=\z@\def\une@rep{\AMCrep@ligne}\AMCrep@init{#1}%
979 \par\begin{center}}%
980 {\AMC@fin@rep\end{center}}
981 \newenvironment{choicescustom}[1][r]{%
982 \AMCrep@count=\z@\def\une@rep{\AMCrep@perso}\AMCrep@init{#1}%
983 \AMCbeginAnswer\ignorespaces}%
984 {\AMC@fin@rep\AMCendAnswer}
985 \newenvironment{tikz-single}[1][r]{
986 \AMCrep@count=\z@\def\une@rep{\AMCrep@tikz}\AMCrep@init{#1}%
987 \begin{tikzpicture}}{\AMC@fin@rep\end{tikzpicture}}
988 \newenvironment{tikz-multi}[1][r]{
989 \AMCrep@count=\z@\def\une@rep{\AMCrep@tikzmat}\AMCrep@init{#1}%
990 \begin{tikzpicture}[remember picture]}{\AMC@fin@rep\end{tikzpicture}}

```

`\AMCrep@bloc` For each of these styles, a corresponding `\AMCrep@xxx{<box>}{<text>}` is defined, which will format the answer with a box given in `<box>` and text `<text>`. `\AMCrep@bloc` is also defined and used in standard formatting when the user wants to put answers inside a L^AT_EX box.

`\AMCrep@tikz`

`\AMCrep@tikzmat`

`\AMCrep@itemize` 991 \newcommand\AMCrep@bloc[2]{\AMC@mem@answer{#1}%

`\AMCrep@ligne`

`\AMCrep@perso`

```

992 \par%
993 \ifAMC@textPos\ vbox\bgroup\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\hbox\bgroup
994 \noindent\begin{minipage}{\linewidth}%
995 \begin{itemize}\item[1] #2\end{itemize}\end{minipage}%
996 \ifAMC@textPos\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\egroup\AMC@tracepos
997 \vspace{\AMCinterBrep}}
998 \newcommand\AMCrep@tikz[5]{\AMC@mem@answer{1}\node[#4] (lab\thecsvrow) at (#3) {#2} node[#5] (box\thecsvrow)
999 \newcommand\AMCrep@tikzmat[5]{\AMC@mem@answer{1}\node[#5] (box\thecsvrow) at (#3) {#1} node[#4] (lab\thecsvrow)
1000 \newcommand\AMCrep@itemize[2]{\AMC@mem@answer{1}\item[1] #2}
1001 \newlength\AMChorizAnswerSep
1002 \setlength\AMChorizAnswerSep{3em plus 4em}
1003 \newlength\AMChorizBoxSep
1004 \setlength\AMChorizBoxSep{1em}
1005 \newcommand\AMCrep@ligne[2]{\AMC@mem@answer{1}}
1006 \ifAMC@textPos%
1007 \mbox{\AMC@tracebox{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}{#1}\hspace*{\AMChorizBoxSep}#2}%
1008 \else%
1009 \mbox{#1\hspace*{\AMChorizBoxSep}#2}%
1010 \fi\hspace{\AMChorizAnswerSep}}
1011 \newcommand\AMCrep@perso[2]{\AMC@mem@answer{1}\AMCanswer{1}{#2}}

```

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

```

1012 \def\AMCbeginAnswer{}
1013 \def\AMCanswer#1#2{#1 #2}
1014 \def\AMCendAnswer{}

```

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

```

\correctchoice
\wrongchoice
1015 \newcommand\correctchoice[2][\global\advance\AMCrep@count\@ne\relax%
1016 \ifAMC@calibration\AMC@amclog{AUTOQCM[REP=\the\AMCrep@count:B]^^J}\fi%
1017 \global\AMCune@bonnettrue%
1018 \AMCload@@reponse{\une@rep{\ifAMC@correc\AMC@box{1}{\AMC@checkedbox}%
1019 \else\AMC@box{1}{\fi}{#2}}{\the\AMCrep@count}\ignorespaces}
1020 \newcommand\wrongchoice[2][\global\advance\AMCrep@count\@ne\relax%
1021 \ifAMC@calibration\AMC@amclog{AUTOQCM[REP=\the\AMCrep@count:M]^^J}\fi%
1022 \AMCload@@reponse{\une@rep{\AMC@box{1}{\fi}{#2}}{\the\AMCrep@count}%
1023 \ignorespaces}

```

4.11.4 Score zones

`\AMCscoreZone` The position of the scores on the annotated answer sheets can be defined in the \LaTeX 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`) prints a marginal note in the left or right margin, depending on current the position (usefull in `multicols` environment).

```

1024 \newcommand{\AMCemptybox}[3]{\%
1025   \sbox0{\wd0=#1\ht0=#2\dp0=#3\relax\box0}}
1026 \newlength\AMC@mn@test
1027 \newlength\AMC@mn@sep\AMC@mn@sep=4mm
1028 \newlength\AMC@mn@leftmargin
1029 \newlength\AMC@mn@rightmargin
1030 \newcommand\AMCmarginNote[1]{\%
1031   \begin{tikzpicture}[remember picture,overlay]\%
1032     \coordinate (here) at (0,0);\%
1033     \pgfextractx{\AMC@mn@test}{\pgfpointdiff{\pgfpointorigin}\%
1034       {\pgfpointanchor{current page}{center}}}\%
1035     \ifodd\thepage\%
1036       \AMC@mn@leftmargin=\oddsidemargin\%
1037       \AMC@mn@rightmargin=\evensidemargin\%
1038     \else
1039       \AMC@mn@leftmargin=\evensidemargin\%
1040       \AMC@mn@rightmargin=\oddsidemargin\%
1041     \fi
1042     \ifdim\AMC@mn@test < 1cm\%
1043       \draw (current page.east |- here)+(-\AMC@mn@rightmargin-1in+\AMC@mn@sep,0pt) node[anchor=text,align=left]{%
1044         \else\%
1045         \draw (current page.west |- here)+(0cm,0pt) node[anchor=text,align=right,text width=\AMC@mn@leftmargin]{%
1046           \fi\%
1047       \end{tikzpicture}}\%
1048 }

```

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)

```

1049 \newcommand{\AMC@sz@box}{\AMCemptybox{\AMC@sz@width}{\AMC@sz@height}{\AMC@sz@depth}}
1050 \%
1051 \newcommand{\AMC@sz@callin@question}{\AMCscoreZone{\AMC@sz@box}}
1052 \%
1053 \newcommand{\AMC@sz@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sz@box}}}
1054 \%
1055 \newcommand{\AMC@sz@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper margin}}
1056 \newcommand{\AMC@sz@callout@margins}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}

```

Let us now set up options handling.

```

1057 \newlength\AMC@sz@width
1058 \newlength\AMC@sz@height
1059 \newlength\AMC@sz@depth
1060 \def\AMC@sz@callout{}

```

```

1061 \def\AMC@sz@callin{}
1062 \define@key{AMC@sz}{width}{\AMC@sz@width=#1}
1063 \define@key{AMC@sz}{height}{\AMC@sz@height=#1}
1064 \define@key{AMC@sz}{depth}{\AMC@sz@depth=#1}
1065 \define@key{AMC@sz}{calloutsideside}{\def\AMC@sz@callout{#1}}
1066 \define@key{AMC@sz}{callinsideside}{\def\AMC@sz@callin{#1}}
1067 \define@choicekey{AMC@sz}{position}{none,question,margin,margins}{%
1068   \ifcsname AMC@sz@callout@#1\endcsname%
1069     \def\AMC@sz@callout{AMC@sz@callout@#1}%
1070   \else%
1071     \def\AMC@sz@callout{}}%
1072 \fi%
1073 \ifcsname AMC@sz@callin@#1\endcsname%
1074   \def\AMC@sz@callin{AMC@sz@callin@#1}%
1075 \else%
1076   \def\AMC@sz@callin{}}%
1077 \fi%
1078 \ifcsname AMC@sz@init@#1\endcsname%
1079   \csname AMC@sz@init@#1\endcsname%
1080 \fi%
1081 }
1082 \newcommand{\AMCsetScoreZone}[1]{\setkeys{AMC@sz}{#1}}
1083 \AMCsetScoreZone{width=1.5em,height=1.5ex,depth=.5ex,position=none}

```

And do the same for \AMCsetScoreZoneAnswerSheet...

```

1084 \newcommand{\AMC@sza@box}{\AMCemptybox{\AMC@sza@width}{\AMC@sza@height}{\AMC@sza@depth}}
1085 %
1086 \newcommand{\AMC@sza@init@none}{}
1087 \newcommand{\AMC@sza@callout@none}{}
1088 \newcommand{\AMC@sza@callin@none}{}
1089 %
1090 \newcommand{\AMC@sza@init@question}{}
1091 \newcommand{\AMC@sza@callout@question}{}
1092 \newcommand{\AMC@sza@callin@question}{\AMCscoreZone{\AMC@sza@box}}
1093 %
1094 \newcommand{\AMC@sza@init@margin}{}
1095 \newcommand{\AMC@sza@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sza@box}}}
1096 \newcommand{\AMC@sza@callin@margin}{}
1097 %
1098 \newcommand{\AMC@sza@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper margin}}
1099 \newcommand{\AMC@sza@callout@margins}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}
1100 \newcommand{\AMC@sza@callin@margins}{}
1101 %
1102 \newlength\AMC@sza@width
1103 \newlength\AMC@sza@height
1104 \newlength\AMC@sza@depth
1105 \def\AMC@sza@callout{}
1106 \def\AMC@sza@callin{}
1107 \define@key{AMC@sza}{width}{\AMC@sza@width=#1}
1108 \define@key{AMC@sza}{height}{\AMC@sza@height=#1}
1109 \define@key{AMC@sza}{depth}{\AMC@sza@depth=#1}

```

```

1110 \define@key{AMCsza}{calloutside}{\def\AMC@sza@callout{#1}}
1111 \define@key{AMCsza}{callinside}{\def\AMC@sza@callin{#1}}
1112 \define@choicekey{AMCsza}{position}{none,question,margin,margins}{%
1113   \ifcsname AMC@sza@callout@#1\endcsname%
1114     \def\AMC@sza@callout{AMC@sza@callout@#1}%
1115   \else%
1116     \def\AMC@sza@callout{}}%
1117 \fi%
1118 \ifcsname AMC@sza@callin@#1\endcsname%
1119   \def\AMC@sza@callin{AMC@sza@callin@#1}%
1120 \else%
1121   \def\AMC@sza@callin{}}%
1122 \fi%
1123 \ifcsname AMC@sza@init@#1\endcsname%
1124   \csname AMC@sza@init@#1\endcsname%
1125 \fi%
1126 }
1127 \newcommand{\AMCsetScoreZoneAnswerSheet}[1]{\setkeys{AMCsza}{#1}}
1128 \AMCsetScoreZoneAnswerSheet{width=1.5em,height=1.5ex,depth=.5ex,position=none}
1129 \newcommand{\AMCnoScoreZone}{\AMCsetScoreZone{position=none}\AMCsetScoreZoneAnswerSheet{position=none}}

```

4.11.5 Formatting questions

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

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

```

1130 \newcounter{AMCquestionaff}
1131 \newcommand{\AMCnumero}[1]{\setcounter{AMCquestionaff}{#1}\addtocounter{AMCquestionaff}{-1}}
1132 \AtBeginDocument{%
1133   \ifx\@skiphyperreftrue\@undefined%
1134     \expandafter\newif\csname if@skiphyperref\endcsname%
1135   \fi%
1136 }
1137 \newcommand\AMC@stepQuestion{\ifAMCquestionNumber\@skiphyperreftrue\refstepcounter{AMCquestionaff}\@skiphyperreftrue}
1138 \newcommand\AMC@qaff{\arabic{AMCquestionaff}}

```

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

```

1139 \def\AMCbeforeQuestion{\ifAMC@qbloc\else\par\noindent\fi}
1140 \def\AMCbeginQuestion#1#2{\noindent\AMC@loc@q{#1}{#2}%
1141   \ifx\@empty\AMC@sz@callin\@empty\hspace*{1em}\fi%
1142 }
1143 \def\multiSymbole{${\clubsuit}$}

```

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

`questionmult`

`variable-single`

`variable-multi`

`questionouverte`

`\ouverte@vs`

Environment `{questionmult}``{\langle key \rangle}` is the same for multiple questions (with none, one or several correct choices).

Environment `{questionmultx}``{\langle key \rangle}` is the same as `questionmult`, but with no use of `\multiSymbole`.

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

```

1144 \ifx\question\undefined\else\let\question\undefined\fi
1145 \def\AMCnobloc{\AMC@qblocfalse}
1146 \def\AMCbloc{\AMC@qbloctrue}
1147 \newenvironment{question}[2][\{%
1148   \def\AMCcurrentenv{question}%
1149   \AMC@stepQuestion%
1150   \global\def\AMCid@name{#2}\AMC@affecte{#2}{\AMCid@quest}%
1151   \ifAMC@calibration\AMCmessage{Q=\the\AMCid@quest}\fi%
1152   \AMCbeforeQuestion%
1153   \ifx\@empty\AMC@sz@callout\@empty\else%
1154     \csname\AMC@sz@callout\endcsname%
1155   \fi%
1156   \AMCtype@multifalse\ifAMC@qbloc\ifAMC@textPos\vbox\bgroup\AMC@tracepos{1}{qtext:#2:\the\AMCid@quest,0}\hbox{
1157     \ifAMC@affichekeys\index{\texttt{#2}}\ifAMC@keyline[\texttt{#2}]\newline\fi\fi%
1158     \AMCbeginQuestion{\ifAMC@affichekeys\ifAMC@ensemble\AMC@qaff\ \fi\ifAMC@keyline\else[\texttt{#2}]\fi\else
1159     \ifx\@empty\AMC@sz@callin\@empty\else%
1160       \csname\AMC@sz@callin\endcsname%
1161     \fi%
1162     \AMCformulaire@dedansfalse\setcounter{AMC@ncase}{0}%
1163     \AMC@mem@openQuestion}%
1164   {\ifAMC@qbloc\end{minipage}\ifAMC@textPos\AMC@tracepos{1}{qtext:\AMCid@name:\the\AMCid@quest,0}\egroup\AMC@t
1165   \newenvironment{questionmult}[1]{%
1166     \AMC@bonnefalse\begin{question}[{\multiSymbole}]{#1}%
1167     \AMCtype@multitrue\ifAMC@calibration%
1168     \AMC@amclog{AUTOQCM[MULT]^^J}\fi}%
1169   {\end{question}}
1170 \newenvironment{variable-single}[2]
1171 {\def\AMCbeginQuestion##1##2{
1172 \begin{questionmult}{#1}\scoring{v=#2}
1173 \begin{tikz-single}[o]}
1174 {\end{tikz-single}
1175 \end{questionmult}}
1176 \newenvironment{variable-multi}[4]
1177 {\def\AMCbeginQuestion##1##2{
1178 \begin{questionmult}{#1}\scoring{v=#4}
1179 \begin{tikz-multi}[o]
1180 \node[#3] (var) at (0,0) {#2};}
1181 {\end{tikz-multi}
1182 \end{questionmult}}
1183 \newenvironment{questionmultx}[1]{%
1184   \begingroup\def\multiSymbole{}\begin{questionmult}{#1}}%
1185 {\end{questionmult}\endgroup}
1186 \newdimen\ouverte@vs
1187 \newenvironment{questionouverte}[1][3cm]{%

```

```

1188 \AMC@stepQuestion%
1189 \AMCtype@multifalse\ouverte@vs=#1%
1190 \ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1191 \AMCbeginQuestion{\AMC@qaff}{}}%
1192 {\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.

```

1193 \newcommand{\explain}[1]{%
1194 \ifAMC@correthead%
1195 \AMCif@env{question}{\par\noindent{\AMC@loc@explain #1}}{\AMC@error@explain}\vspace{1ex}%
1196 \else%
1197 \AMCif@env{question}{}{\AMC@error@explain}%
1198 \fi%
1199 }

```

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.

```

1200 \def\scoring#1{\ifAMC@calibration\AMC@amclog{AUTOQCM[B=#1]^^J}\fi}
1201 \def\scoringDefaultS#1{\ifAMC@calibration\AMC@amclog{AUTOQCM[BDS=#1]^^J}\fi}
1202 \def\scoringDefaultM#1{\ifAMC@calibration\AMC@amclog{AUTOQCM[BDM=#1]^^J}\fi}
1203 \def\QuestionIndicative{\ifAMC@calibration\AMC@amclog{AUTOQCM[INDIC]^^J}\fi}

```

4.13 Numerical data

4.13.1 Codes

`\AMCcodeGrid`
`\AMCcodeGridInt`

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.

```

1204 \newcount\AMC@chiffres
1205 \newdimen\AMCcodeHspace\AMCcodeHspace=.5em
1206 \newdimen\AMCcodeVspace\AMCcodeVspace=.5em
1207 \newcommand\AMCcodeID@squarebrackets[2]{#1[#2]}
1208 \newcommand\AMCcodeID@dot[2]{#1.#2}
1209 \newcommand\AMCcodeID@[1]{%
1210   \expandafter\def\expandafter\AMCcodeID\expandafter{\csname AMCcodeID@#1\endcsname}%
1211 }
1212 \AMCcodeID@[squarebrackets]
```

```

1213 \ExplSyntaxOn
1214
1215 \clist_new:N \amc_code_descr_clist
1216 \seq_new:N \amc_code_digits_seq
1217 \int_new:N \amc_code_digit_n_int
1218 \bool_new:N \amc_code_vertical_bool
1219 \bool_new:N \amc_code_top_bool
1220
1221 \cs_new:Npn \amc_code_init:N #1 {
1222   \def\AMCbeginQuestion##1##2{}
1223   \def\AMCbeforeQuestion{}
1224   \AMCnoScoreZone
1225   \AMCquestionNumberfalse
1226   \setlength{\parindent}{0pt}
1227   \AMCnobloc
1228   \int_set:Nn \amc_code_digit_n_int { \clist_count:N #1 }
1229 }
1230
1231 \cs_new:Nn \amc_code_digit_init: {
1232   \QuestionIndicative
1233   \global\AMCrep@count=\z@
1234 }
1235
1236 \cs_new:Npn \amc_code_digit:n #1 {
1237   \global\advance\AMCrep@count\@ne\relax
1238   \ifAMC@calibration\AMC@amclog{AUTOQCM[ REP = \the\AMCrep@count : M ]^^J}\fi
1239   \hbox{\AMC@keyBox@{#1}{1}{case : \AMCid@name : \the\AMCid@quest , \the\AMCrep@count}}
1240   \bool_if:NTF \amc_code_vertical_bool {
1241     \vspace{\AMCcodeVspace}
1242   }{
1243     \hspace{\AMCcodeHspace}
1244   }
1245 }
1246
1247 \keys_define:nn { amccode } {
1248   vertical .bool_set:N = \amc_code_vertical_bool,
1249   vertical .initial:n = { true },
1250   vertical .default:n = { true },
1251   v .code:n = { \bool_set_true:N \amc_code_vertical_bool },
1252   h .code:n = { \bool_set_false:N \amc_code_vertical_bool },
1253   top .bool_set:N = \amc_code_top_bool,
1254   top .initial:n = { false },
1255   top .default:n = { true }
1256 }
1257
1258 \cs_new:Npn \amc_code_generate:nNn #1#2#3 {
1259   { \keys_set:nn { amccode } { #3 }
1260     \amc_code_init:N #2
1261     \clist_map_inline:Nn #2 { % iterates over 'digits'
1262       \begin{question}\AMCcodeID{#1}{ \int_use:N \amc_code_digit_n_int }}

```

```

1263 \amc_code_digit_init:
1264 \seq_set_split:Nnn \amc_code_digits_seq {} { ##1 }
1265 \bool_if:NTF \amc_code_vertical_bool {
1266   \hspace{0pt}
1267   \bool_if:NTF \amc_code_top_bool { \vtop } { \vbox }
1268   \bgroup
1269   }{
1270   \hbox\bgroup
1271   }
1272   \seq_map_inline:Nn \amc_code_digits_seq {
1273     % iterates over available characters for 'digit'
1274     \amc_code_digit:n { #####1 }
1275   }
1276   \bool_if:NTF \amc_code_vertical_bool {
1277     \vspace{-\AMCcodeVspace}\egroup
1278     \hspace{\AMCcodeHspace}
1279   }{
1280     \egroup\vspace{\AMCcodeVspace}
1281     \par
1282   }
1283   \end{question}
1284   \int_decr:N \amc_code_digit_n_int
1285 }
1286 }
1287 }
1288
1289 \cs_new:Npn \amc_code_generate:nnn #1#2#3 {
1290   \clist_set:Nn \amc_code_descr_clist { #2 }
1291   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1292 }
1293 \cs_generate_variant:Nn \amc_code_generate:nnn { xxx }
1294 \newcommand{\AMCcodeGrid}[3][[]]{
1295   \amc_code_generate:xxx { #2 } { #3 } { #1 }
1296 }
1297
1298 \cs_new:Npn \amc_code_generate_integer:nnn #1#2#3 {
1299   \clist_clear:N \amc_code_descr_clist
1300   \prg_replicate:nn { #2 } { \clist_put_right:Nn \amc_code_descr_clist { 0123456789 } }
1301   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1302 }
1303 \cs_generate_variant:Nn \amc_code_generate_integer:nnn { xxx }
1304 \newcommand{\AMCcodeGridInt}[3][[]]{
1305   \amc_code_generate_integer:xxx { #2 } { #3 } { #1 }
1306 }
1307
1308 \cs_new:Npn \amc_code_generate_integer_v:nn #1#2 {
1309   \amc_code_generate_integer:nnn { #1 } { #2 } { v }
1310 }
1311 \cs_new:Npn \amc_code_generate_integer_h:nn #1#2 {
1312   \amc_code_generate_integer:nnn { #1 } { #2 } { h }

```

```

1313 }
1314 \cs_generate_variant:Nn \amc_code_generate_integer_v:nn { xx }
1315 \cs_generate_variant:Nn \amc_code_generate_integer_h:nn { xx }
1316 \cs_new_eq:NN \AMCcode \amc_code_generate_integer_v:xx
1317 \cs_new_eq:NN \AMCcodeH \amc_code_generate_integer_h:xx
1318
1319 \ExplSyntaxOff

```

4.13.2 Numerical questions

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

As an example,

```

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

```

produces (in correction mode):

Question 3 What is the value of 7×5 ?	
<input type="checkbox"/> 0 <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input checked="" type="checkbox"/> 3 <input type="checkbox"/> 4 <input type="checkbox"/> 5 <input type="checkbox"/> 6 <input type="checkbox"/> 7 <input type="checkbox"/> 8 <input type="checkbox"/> 9	<input type="checkbox"/> 0 <input type="checkbox"/> 1 <input type="checkbox"/> 2 <input type="checkbox"/> 3 <input type="checkbox"/> 4 <input checked="" type="checkbox"/> 5 <input type="checkbox"/> 6 <input type="checkbox"/> 7 <input type="checkbox"/> 8 <input type="checkbox"/> 9

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.

```

1320 \newcommand\AMCformatChoices[4]{%
1321   \global\AMCrep@count=\z@%
1322   \AMC@if@separate@question{%
1323     \AMC@mem@add{\global\AMCrep@count=\z@%
1324               #1{#3}{#4}}%
1325   }%
1326   \ifAMC@ensemble%
1327     #2{#3}{#4}%
1328     \AMC@amclog{AUTOQCM[QPART]^^J}%
1329   \else%
1330     #1{#3}{#4}%
1331   \fi%
1332 }

```

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 $\langle x \rangle$. 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}\langle x \rangle\langle n \text{ digits} \rangle\langle base \rangle` rounds the floating point number $\langle x \rangle$ and populates the comma separated list $\langle clist \rangle$ with its $\langle n \text{ digits} \rangle$ digits in base $\langle base \rangle$. An error is issued if $\langle x \rangle$ would have required more digits.

```
1333 \ExplSyntaxOn
1334
1335 \cs_generate_variant:Nn \tl_replace_once:Nnn { Nxn }
1336
1337 \tl_new:N \amc_ee_tl
1338 \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.

```
1339 \group_begin:
1340 \char_set_catcode_other:N E
1341 \tex_lowercase:D
1342 {
1343   \cs_new:Npn \amc_read_scientific:NNn #1 #2 #3 {
1344     \tl_set:Nn \amc_ee_tl { #3 }
1345     \tl_replace_once:Nxn \amc_ee_tl { E } { e }
1346     \seq_set_split:NnV \amc_ee_seq e \amc_ee_tl
1347     \fp_set:Nn #1 { \seq_item:Nn \amc_ee_seq 1 }
1348     \int_set:Nn #2 { \seq_item:Nn \amc_ee_seq 2 }
1349   }
1350 }
1351 \group_end:
1352
1353 \cs_generate_variant:Nn \amc_read_scientific:NNn { NNf, NNx }
1354
1355 \fp_new:N \amc_fulls_fp
1356 \cs_new:Npn \amc_fp_decompose:NNn #1 #2 #3 {
1357   \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`.

```
1358   \amc_read_scientific:NNx #1 #2
1359   { \fp_to_scientific:N \amc_fulls_fp e 0 }
1360 }
1361 \cs_generate_variant:Nn \amc_fp_decompose:NNn { NNx }
1362
1363 \fp_new:N \amc_num_mantissa_fp
1364 \int_new:N \amc_num_exponent_int
1365 \cs_new:Npn \amc_fp_n_significant_digits:Nnn #1 #2 #3 {
1366   \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_exponent_int
1367   { #2 }
1368   \fp_set:Nn #1
1369   { round(\amc_num_mantissa_fp * 10^((#3)-1)) }
1370   \fp_compare:nTF { abs(#1) >= 10^(#3) }
1371   {
```

```

1372     \fp_set:Nn #1 { #1 / 10 }
1373   } { }
1374 }
1375
1376 \fp_new:N \amc_num_nsig_fp
1377 \cs_new:Npn \amc_fp_show_n_significant_digits:nn #1 #2 {
1378   \amc_fp_n_significant_digits:Nnn \amc_num_nsig_fp { #1 } { #2 }
1379 }
1380 \cs_new_eq:NN \AMCsignificantDigits \amc_fp_show_n_significant_digits:nn
1381
1382 \cs_new:Npn \amc_fp_show_significant_digits: {
1383   \fp_use:N \amc_num_nsig_fp
1384 }
1385 \cs_new_eq:NN \AMCshowSignificantDigits \amc_fp_show_significant_digits:
1386
1387 \cs_new:Npn \amc_fp_n_digits:Nnn #1 #2 #3 {
1388   \fp_set:Nn #1
1389     { round((#2) * 10^(#3)) }
1390 }
1391
1392 \int_new:N \amc_todigits_int
1393 \cs_new:Npn \amc_fp_to_digits:Nnnn #1 #2 #3 #4 {
1394   \clist_clear:N #1
1395   \int_set:Nn \amc_todigits_int { \fp_eval:n { abs(round(#2)) } }
1396   \prg_replicate:nn { #3 } {
1397     \clist_put_left:Nx #1 { \int_mod:nn \amc_todigits_int { #4 } }
1398     \int_set:Nn \amc_todigits_int
1399       { \int_div_truncate:nn \amc_todigits_int { #4 } }
1400   }
1401   \int_compare:nNnTF \amc_todigits_int = 0 { } {
1402     \message{^^J!~Error:~number~too~large,
1403       ~some~digits~will~be~discarded^^J}
1404   }
1405 }
1406
1407 \cs_new:Npn \amc_invalid_digits:Nn #1 #2 {
1408   \clist_clear:N #1
1409   \prg_replicate:nn { #2 } { \clist_put_left:Nx #1 { -1 } }
1410 }
1411
1412 \cs_new:Npn \amc_get_fp_sign:Nn #1 #2 {
1413   \fp_compare:nNnTF #2 < 0 {
1414     \int_set:Nn #1 { -1 }
1415   }{
1416     \fp_compare:nNnTF #2 > 0 {
1417       \int_set:Nn #1 { 1 }
1418     }{
1419       \int_set:Nn #1 { 0 }
1420     }
1421   }

```



```

1422 }
1423
1424 \cs_new:Npn \amc_get_int_sign:Nn #1 #2 {
1425   \int_compare:nNnTF #2 < 0 {
1426     \int_set:Nn #1 { -1 }
1427   }{
1428     \int_compare:nNnTF #2 > 0 {
1429       \int_set:Nn #1 { 1 }
1430     }{
1431       \int_set:Nn #1 { 0 }
1432     }
1433   }
1434 }
1435
1436 \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:

```

1437 \def\AMCnntextGoto{}
1438 \def\AMCnntextVHead#1{\emph{b#1}}
1439 \newdimen\AMCnumeric@Hspace\AMCnumeric@Hspace=.5em
1440 \newdimen\AMCnumeric@Vspace\AMCnumeric@Vspace=1ex
1441 \ExplSyntaxOn
1442
1443 \keys_define:nn { amcnumeric } {
1444   Tsign .code:n = {\def\AMCnntextSign{#1}},
1445   Tsign .initial:n = {},
1446   Tpoint .code:n = {\def\AMCdecimalPoint{#1}},
1447   Tpoint .initial:n = { \raisebox{1ex}{\bf .} },
1448   Texponent .code:n = {\def\AMCexponent{#1}},
1449   Texponent .initial:n = { $\times 10^{\text{asciicircum}} $ },
1450   vspace .code:n = {\AMCnumeric@Vspace=#1},
1451   hspace .code:n = {\AMCnumeric@Hspace=#1},
1452   bordercol .code:n = {\def\AMCncol@Border{#1}},
1453   bordercol .initial:n = { lightgray },
1454   borderwidth .code:n = {\def\AMCncol@BorderWidth{#1}},
1455   borderwidth .initial:n = { 1mm },
1456   backgroundcol .code:n = {\def\AMCncol@Background{#1}},
1457   backgroundcol .initial:n = { white },
1458   digits .int_set:N = \amc_num_ndigits_int,
1459   digits .initial:n = { 3 },
1460   decimals .int_set:N = \amc_num_decd_int,
1461   decimals .initial:n = { 0 },
1462   exponent .int_set:N = \amc_num_expo_int,
1463   exponent .initial:n = { 0 },
1464   base .int_set:N = \amc_num_base_int,
1465   base .initial:n = { 10 },
1466   sign .bool_set:N = \amc_num_sign_bool,

```

```

1467 sign .initial:n = { true },
1468 sign .default:n = { true },
1469 exposign .bool_set:N = \amc_num_exposign_bool,
1470 exposign .initial:n = { true },
1471 exposign .default:n = { true },
1472 strict .bool_set:N = \amc_num_strict_bool,
1473 strict .initial:n = { false },
1474 strict .default:n = { true },
1475 scoring .bool_set:N = \amc_num_scoring_bool,
1476 scoring .initial:n = { true },
1477 scoring .default:n = { true },
1478 vertical .bool_set:N = \amc_num_vertical_bool,
1479 vertical .initial:n = { false },
1480 vertical .default:n = { true },
1481 expoververtical .bool_set:N = \amc_num_expoververtical_bool,
1482 expoververtical .initial:n = { false },
1483 expoververtical .default:n = { true },
1484 reverse .bool_set:N = \amc_num_reverse_bool,
1485 reverse .initial:n = { false },
1486 reverse .default:n = { true },
1487 vhead .bool_set:N = \amc_num_vhead_bool,
1488 vhead .initial:n = { false },
1489 vhead .default:n = { true },
1490 nozero .bool_set:N = \amc_num_nozero_bool,
1491 nozero .initial:n = { false },
1492 nozero .default:n = { true },
1493 significant .bool_set:N = \amc_num_significant_bool,
1494 significant .initial:n = { false },
1495 significant .default:n = { true },
1496 scoreexact .code:n = {\def\AMC@numeric@scoreexact{#1}},
1497 scoreexact .initial:n = { 2 },
1498 scoreapprox .code:n = {\def\AMC@numeric@scoreapprox{#1}},
1499 scoreapprox .initial:n = { 1 },
1500 scorewrong .code:n = {\def\AMC@numeric@scorewrong{#1}},
1501 scorewrong .initial:n = { 0 },
1502 exact .int_set:N = \amc_num_exact_int,
1503 exact .initial:n = { 0 },
1504 approx .int_set:N = \amc_num_approx_int,
1505 approx .initial:n = { 0 },
1506 keepas .code:n = {\def\AMC@numeric@keepas{#1}},
1507 keepas .initial:n = {},
1508 alsocorrect .code:n = {\def\AMC@numeric@alsocorrect{#1}},
1509 alsocorrect .initial:n = {}
1510 }
1511
1512 \cs_new:Npn \amc_num_setopts #1 {
1513   \keys_set:nn { amcnumeric } { #1 }
1514 }
1515
1516 \cs_new_eq:NN \AMCnumericOpts \amc_num_setopts

```

1517

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.

```

1518 \cs_new:Npn \amc_num_char:nn #1 #2 {
1519   \global\advance\AMCrep@count\one\relax
1520   \AMC@amclog{AUTOQCM[REP= \the\AMCrep@count :
1521     \ifx#2\AMC@checkbox B\else M\fi ]^^J}
1522   \ifAMC@correc
1523     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :
1524       \the\AMCid@quest , \the\AMCrep@count}
1525   \else
1526     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :
1527       \the\AMCid@quest , \the\AMCrep@count}
1528   \fi
1529 }
```

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

```

1530 \int_new:N \amc_num_digit_value_int
1531 \tl_new:N \amc_num_digit_value_tl
1532 \cs_new:Npn \amc_num_digit_box:nn #1 #2 {
1533   \int_set:Nn \amc_num_digit_value_int { #1 }
1534   \tl_set:Nn \amc_num_digit_value_tl {
1535     \int_compare:nNnTF { \amc_num_digit_value_int } < { 10 }
1536       { \int_to_arabic:n { \amc_num_digit_value_int } }
1537       { \int_to_Alph:n { \amc_num_digit_value_int - 9 } }
1538   }
1539   \int_compare:nNnTF { #1 } = { #2 } {
1540     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1541       {\AMC@checkbox}
1542   } {
1543     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1544       {}
1545   }
1546 }
```

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

```

1547 \cs_new:Npn \amc_num_sign_boxes:Nn #1 #2 {
1548   \int_case:nn { #1 } {
1549     { -1 } {
1550       \hbox{\amc_num_char:nn{ $+ $ }{ }}
1551       \vspace{\AMCnumeric@Vspace}
1552       \AMC@amclog{AUTOQCM[B=set. sign #2 =1]^^J}
1553       \hbox{\amc_num_char:nn{ $- $ }{\AMC@checkbox}}
1554       \AMC@amclog{AUTOQCM[B=set. sign #2 =-1]^^J}
1555     }
1556     { 1 } {
```

```

1557 \hbox{\amc_num_char:nn{${$}}{\AMC@checkbox}}
1558 \vspace{\AMCnumeric@Vspace}
1559 \AMC@amclog{AUTOQCM[B=set. sign #2 =1]^^J}
1560 \hbox{\amc_num_char:nn{$-$}}{\AMC@checkbox}}
1561 \AMC@amclog{AUTOQCM[B=set. sign #2 =-1]^^J}
1562 }
1563 { 0 } {
1564 \hbox{\amc_num_char:nn{${$}}{\AMC@checkbox}}
1565 \vspace{\AMCnumeric@Vspace}
1566 \AMC@amclog{AUTOQCM[B=set. sign #2 =1]^^J}
1567 \hbox{\amc_num_char:nn{$-$}}{\AMC@checkbox}}
1568 \AMC@amclog{AUTOQCM[B=set. sign #2 =-1]^^J}
1569 }
1570 }
1571 }

```

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

```

1572 \cs_new:Npn \amc_num_digit_boxes_h:nnn #1 #2 #3 {
1573 \int_step_inline:nnnn
1574 { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1575 { 1 } { #3 - 1 } {
1576 \amc_num_digit_box:nn { ##1 }{ #2 }
1577 \AMC@amclog{AUTOQCM[B= set. #1 = ##1 ]^^J}
1578 \hspace{\AMCnumeric@Hspace}
1579 }
1580 \hspace{-\AMCnumeric@Hspace}
1581 }
1582
1583 \cs_new:Npn \amc_num_digit_boxes_v:nnn #1 #2 #3 {
1584 \int_step_inline:nnnn
1585 { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1586 { 1 } { #3 - 1 } {
1587 \vbox{\hbox{
1588 \amc_num_digit_box:nn { ##1 }{ #2 }
1589 }}
1590 \AMC@amclog{AUTOQCM[B= set. #1 = ##1 ]^^J}
1591 \int_compare:nNnTF { ##1 } < { #3 - 1 } {
1592 \vspace{\AMCnumeric@Vspace}
1593 } {}
1594 }
1595 }
1596
1597 \int_new:N \amc_num_first_digit_int
1598 \cs_new:Npn \amc_num_digit_boxes_vr:nnn #1 #2 #3 {
1599 \int_set:Nn \amc_num_first_digit_int
1600 { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1601 \int_step_inline:nnnn { #3 - 1 } { -1 }

```

```

1602 \amc_num_first_digit_int {
1603   \vbox{\hbox{
1604     \amc_num_digit_box:nn { ##1 } { #2 }
1605   }}
1606   \AMC@amclog{AUTOQCM[B= set. #1 = ##1 ]^^J}
1607   \int_compare:nNnTF { ##1 } > \amc_num_first_digit_int {
1608     \vspace{\AMCnumeric@Vspace}
1609   } {}
1610 }
1611 }

```

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

```

1612 \cs_new:Npn \amc_num_integer_boxes_v:Nnn #1 #2 #3 {
  begin a loop over all digits,
1613   \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1614   \clist_map_inline:Nn #1 {
    place the decimal point if necessary,
1615     \int_compare:nNnTF \amc_num_digit_int = { #3 } {
1616       \hbox{ \AMCdecimalPoint } \hspace{\AMCnumeric@Hspace}
1617     } {}
    draw the box for this digit,
1618     \hbox{\vbox{
1619       \bool_if:NTF \amc_num_vhead_bool {
1620         \vbox{\hbox{\AMCtextVHead{ \int_eval:n
1621           { \amc_num_digit_int - 1 } }}}
1622         \vspace{\AMCnumeric@Vspace}
1623       } {}
1624       \bool_if:NTF \amc_num_reverse_bool {
1625         \amc_num_digit_boxes_vr:nnn { #2
1626           \int_to_Alph:n \amc_num_digit_int }
1627         { ##1 } { \amc_num_base_int }
1628       } {
1629         \amc_num_digit_boxes_v:nnn { #2
1630           \int_to_Alph:n \amc_num_digit_int }
1631         { ##1 } { \amc_num_base_int }
1632       }
1633     }}
  }

```

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

```

1634   \int_compare:nNnTF \amc_num_digit_int > 1 {
1635     \hspace{\AMCnumeric@Hspace}
1636   } {}
1637   \int_decr:N \amc_num_digit_int
1638 }
1639 }
1640

```

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

```

1641
1642 \cs_new:Npn \amc_num_integer_boxes_h:Nnn #1 #2 #3 {
1643   \vbox{
1644     \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1645     \clist_map_inline:Nn #1 {
1646       \int_compare:nNnTF
1647         \amc_num_digit_int = { #3 } {
1648           \hbox{ \AMCdecimalPoint }
1649         } { }
1650       \hbox{
1651         \amc_num_digit_boxes_h:nnn { #2
1652           \int_to_Alph:n \amc_num_digit_int }
1653         { ##1 } \amc_num_base_int
1654       }
1655       \int_compare:nNnTF \amc_num_digit_int > 1 {
1656         \vspace{\AMCnumeric@Vspace}
1657       } { }
1658       \int_decr:N \amc_num_digit_int
1659     }
1660   }
1661

```

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

```

1662
1663 \cs_new:Npn \amc_num_integer_boxes:NnnNN #1 #2 #3 #4 #5 {
1664   \hbox{
1665     \bool_if:NTF { #4 } {
1666       \vbox{
1667         \ifx\AMCtextSign\@empty\@empty\else
1668           \hbox{\AMCtextSign}\vspace{\AMCnumeric@Vspace}\fi
1669           \amc_num_sign_boxes:Nn { #5 } { #2 }
1670         }
1671         \hspace{.5em}
1672         \vrule
1673         \hspace{.5em}
1674       } { }
1675       \hbox{
1676         \bool_if:NTF \amc_num_vertical_bool
1677         \amc_num_integer_boxes_v:Nnn \amc_num_integer_boxes_h:Nnn
1678         #1 { #2 } { #3 }
1679       }
1680     }
1681   }
1682

```

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.

```

1683
1684 \cs_new:Npn \amc_num_build_integer_scoring:Nnnn #1 #2 #3 #4 {

```

```

1685 \tl_clear:N #1
1686 \int_set_eq:NN \amc_num_digit_int { #4 }
1687 \int_while_do:nNnn \amc_num_digit_int > 0 {
1688   \bool_if:NTF \amc_num_strict_bool {
1689     \AMC@amclog{AUTOQCM[B=requires. #3
1690       \int_to_Alph:n \amc_num_digit_int = 1]^^J}
1691   } {
1692     \AMC@amclog{AUTOQCM[B=default. #3
1693       \int_to_Alph:n \amc_num_digit_int = 0]^^J}
1694   }
1695   \int_compare:nNnTF \amc_num_digit_int = #4 { } {
1696     \tl_put_left:Nn #1 { ( }
1697     \tl_put_right:Nx #1 { ) *
1698     \int_use:N \amc_num_base_int + }
1699   }
1700   \tl_put_right:Nx #1
1701   { #3 \int_to_Alph:n \amc_num_digit_int }
1702   \int_decr:N \amc_num_digit_int
1703 }
1704 \tl_put_left:Nn #1 { ( }
1705 \tl_put_right:Nn #1 { ) }
1706 \bool_if:NTF { #2 } {
1707   \bool_if:NTF \amc_num_strict_bool {
1708     \AMC@amclog{AUTOQCM[B=requires. sign #3 =1]^^J}
1709   } {
1710     \AMC@amclog{AUTOQCM[B=default. sign #3 =1]^^J}
1711   }
1712   \tl_put_right:Nx #1 { * ( sign #3 ) }
1713 } { }
1714 }
1715

```

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

```

1716
1717 \fp_new:N \amc_num_result_fp
1718 \fp_new:N \amc_num_correct_fp
1719 \clist_new:N \amc_num_digits_clist
1720 \clist_new:N \amc_num_expo_digits_clist
1721 \int_new:N \amc_num_digit_int
1722 \int_new:N \amc_num_sign_int
1723 \int_new:N \amc_num_expo_sign_int
1724 \tl_new:N \amc_num_compute_tl
1725 \tl_new:N \amc_num_expo_tl
1726 \int_new:N \amc_num_correct_expo_int
1727
1728 \cs_new:Npn \amc_numeric_show:nn #1 #2 {

```

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

```

1729 \ifAMC@ensemble\ifAMCformulaire@dedans
1730   \AMC@amclog{AUTOQCM[Q=\the\AMCid@quest]^^J}
1731 \fi\fi

```

Then we parse the options from $\langle opts \rangle$:

```

1732 { \keys_set:nn { amcnumeric } { #2 }
1733   \bool_if:nTF { \bool_if_p:N\amc_num_significant_bool
1734     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1735     \message{^^J!~AMCnumeric~Error:~significant=true~can't~be~used~with~base!=10.^^J}
1736   } {}
1737   \bool_if:nTF { \int_compare_p:n { \amc_num_expo_int != 0 }
1738     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1739     \message{^^J!~AMCnumeric~Error:~scientific~notation~can't~be~used~with~base!=10.^^J}
1740   } {}

```

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

```

1741 \ifx\@empty#1\@empty
1742   \fp_set:Nn \amc_num_correct_fp { 0 }
1743   \fp_set:Nn \amc_num_mantissa_fp { 0 }
1744   \int_set:Nn \amc_num_correct_expo_int { 0 }
1745 \else
1746   \bool_if:NTF \amc_num_significant_bool {
1747     \amc_fp_n_significant_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_ndigits_int
1748   } {
1749     \int_compare:nNnTF \amc_num_expo_int > 0 {
1750       \amc_fp_decompose:Nnn \amc_num_mantissa_fp \amc_num_correct_expo_int { #1 }
1751       \int_compare:nNnTF { \amc_num_ndigits_int - \amc_num_decd_int } > 1 {
1752         \fp_set:Nn \amc_num_mantissa_fp {
1753           \amc_num_mantissa_fp * 10^( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1754         }
1755         \int_set:Nn \amc_num_correct_expo_int {
1756           \amc_num_correct_expo_int - ( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1757         }
1758       } {}
1759       \amc_fp_n_digits:Nnn \amc_num_correct_fp \amc_num_mantissa_fp \amc_num_decd_int
1760     } {
1761       \amc_fp_n_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_decd_int
1762     }
1763   }
1764 \fi

```

Now extracts the required digits:

```

1765 \ifx\@empty#1\@empty
1766   \amc_invalid_digits:Nn \amc_num_digits_clist \amc_num_ndigits_int
1767   \amc_invalid_digits:Nn \amc_num_expo_digits_clist \amc_num_expo_int
1768   \int_set:Nn \amc_num_sign_int { 0 }
1769   \int_set:Nn \amc_num_expo_sign_int { 0 }
1770 \else
1771   \amc_fp_to_digits:Nnnn \amc_num_digits_clist \amc_num_correct_fp
1772     \amc_num_ndigits_int \amc_num_base_int
1773   \amc_get_fp_sign:Nn \amc_num_sign_int \amc_num_correct_fp
1774   \int_compare:nNnTF \amc_num_expo_int > 0 {
1775     \amc_fp_to_digits:Nnnn \amc_num_expo_digits_clist \amc_num_correct_expo_int
1776       \amc_num_expo_int \amc_num_base_int

```



```

1777     \amc_get_int_sign:Nn \amc_num_expo_sign_int \amc_num_correct_expo_int
1778   } {}
1779   \fi

```

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

```

1780   \fp_set:Nn \amc_num_result_fp { #1 }
1781   \AMC@amclog{AUTOQCM[B=numval=\fp_to_scientific:N \amc_num_result_fp ,
1782     numex=\int_use:N \amc_num_exact_int,
1783     numapp=\int_use:N \amc_num_approx_int,
1784     numsex=\AMC@numeric@scoreexact,
1785     numsapp=\AMC@numeric@scoreapprox
1786   ]}
1787   \bool_if:NTF \amc_num_scoring_bool {
1788     \AMC@amclog{AUTOQCM[B=haut=mz=,
1789       formula=(Vdifference <= \int_use:N \amc_num_exact_int ?
1790         \AMC@numeric@scoreexact :
1791         \int_compare:nNnTF \amc_num_approx_int = 0 {
1792           \AMC@numeric@scorewrong
1793         } {
1794           (Vdifference <= \int_use:N\amc_num_approx_int ?
1795             \AMC@numeric@scoreapprox : \AMC@numeric@scorewrong)
1796         }
1797       )]^^J}
1798   } {}
1799   \amc_num_build_integer_scoring:Nnnn
1800   \amc_num_compute_tl \amc_num_sign_bool { digit } \amc_num_ndigits_int
1801   \int_compare:nNnTF \amc_num_expo_int > 0 {
1802     \amc_num_build_integer_scoring:Nnnn
1803     \amc_num_expo_tl \amc_num_exposign_bool { expo } \amc_num_expo_int
1804     \AMC@amclog{AUTOQCM[B= set. intE = \amc_num_expo_tl ]^^J}
1805   } {}
1806   \AMC@amclog{AUTOQCM[B= set.intV = \fp_to_int:N\amc_num_correct_fp ,
1807     set.intXX = \amc_num_compute_tl ]^^J}
1808   \int_compare:nNnTF \amc_num_expo_int > 0 {
1809     \AMC@amclog{AUTOQCM[B= set.intX = intXX * \int_use:N\amc_num_base_int ** ( intE - (\int_use:N\amc_num_co
1810   ){
1811     \AMC@amclog{AUTOQCM[B= set.intX = intXX]^^J}
1812   }
1813   \int_compare:nNnTF \amc_num_expo_int > 0 {
1814     \AMC@amclog{AUTOQCM[B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (intE - \int_use:N\amc_num_d
1815   ){
1816     \AMC@amclog{AUTOQCM[B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (- \int_use:N\amc_num_decd_i
1817   }
1818   \ifx\@empty\AMC@numeric@keepas\@empty\else
1819     \AMC@amclog{AUTOQCM[B= setglobal.\AMC@numeric@keepas = valueX ]^^J}
1820   \fi
1821   \ifx\@empty#1\@empty
1822     \bool_if:NTF \amc_num_significant_bool {
1823       \AMC@amclog{AUTOQCM[B=set.Vdifference=0]^^J}
1824     }{

```

```

1825 \ifx\@empty\AMC@numeric@alsocorrect\@empty
1826 \AMC@amclog{AUTOQCM[B=set.Vdifference=0]^^J}
1827 \else
1828 \AMC@amclog{AUTOQCM[B="set.Vdifference =
1829 amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\amc_num_e
1830 "]"^^J}
1831 \fi
1832 }
1833 \else
1834 \bool_if:NTF \amc_num_significant_bool {
1835 \AMC@amclog{AUTOQCM[B=set.Vdifference="min( abs((intV)-(intX)) ,
1836 abs(\int_use:N\amc_num_base_int * (intV) - (intX)) ,
1837 abs((intV) - \int_use:N\amc_num_base_int * (intX)) )"^^J}
1838 } {
1839 \ifx\@empty\AMC@numeric@alsocorrect\@empty
1840 \AMC@amclog{AUTOQCM[B=set.Vdifference=abs((intV)-(intX))]^^J}
1841 \else
1842 \AMC@amclog{AUTOQCM[B="set.Vdifference =
1843 min( amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\amc_num
1844 abs((intV)-(intX)) )" ]^^J}
1845 \fi
1846 }
1847 \fi

```

Begin now with the frame around all the boxes:

```

1848 \ifAMC@extractOnly\else
1849 \vspace{1.5ex}\par{
1850 \fboxrule=\AMCncol@BorderWidth
1851 \fcolorbox{\AMCncol@Border}{\AMCncol@Background}{
1852 \bool_if:NTF \amc_num_expovetical_bool {
1853 \hbox{\vbox{
1854 \vbox{\amc_num_integer_boxes:NnnNN
1855 \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1856 \amc_num_sign_int}
1857 \int_compare:nNnTF \amc_num_expo_int > 0 {
1858 \vspace{\AMCnumeric@Vspace}
1859 \vbox{\hbox{\AMCexponent}}
1860 \vspace{\AMCnumeric@Vspace}
1861 \vbox{\amc_num_integer_boxes:NnnNN
1862 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
1863 \amc_num_expo_sign_int}
1864 } {}
1865 }}
1866 } {
1867 \amc_num_integer_boxes:NnnNN
1868 \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1869 \amc_num_sign_int
1870 \int_compare:nNnTF \amc_num_expo_int > 0 {
1871 \hspace{\AMCnumeric@Hspace}\AMCexponent\hspace{\AMCnumeric@Hspace}
1872 \amc_num_integer_boxes:NnnNN
1873 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool

```

```

1874      \amc_num_expo_sign_int
1875      } {}
1876    }
1877  }
1878 }
1879 \fi

```

And tell AMC that we finished with this question:

```

1880 \ifAMC@ensemble\else\vspace{1.5ex}\par\fi
1881 \ifAMC@ensemble\ifAMCformulaire@dedans
1882   \AMC@amclog{AUTOQCM[FQ]^^J}
1883 \fi\fi
1884 }
1885 }
1886
1887 \cs_new_eq:NN \AMCnumericShow \amc_numeric_show:nn
1888

```

`\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.

```

1889 \cs_new:Npn \amc_numeric_hide:nn #1 #2 {
1890   \keys_set:nn { amcnumeric } { #2 }
1891   \AMCnxtxtGoto
1892   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi
1893 }
1894
1895 \cs_new_eq:NN \AMCnumericHide \amc_numeric_hide:nn
1896
1897 \ExplSyntaxOff
1898 \def\AMCnumericChoicesPlain{%
1899   \AMC@if@separate@question{\AMC@mem@category{numeric}}%
1900   \AMCformatChoices{\AMCnumericShow}{\AMCnumericHide}%
1901 }

```

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

```

1902 \ExplSyntaxOn
1903
1904 \cs_new:Npn \amc_numeric_choices:nn #1#2 {
1905   \AMCnumericChoicesPlain{#1}{#2}
1906 }
1907 \cs_generate_variant:Nn \amc_numeric_choices:nn { xn }
1908 \cs_new_eq:NN \AMCnumericChoices \amc_numeric_choices:xn
1909
1910 \ExplSyntaxOff

```

4.13.3 Intervals

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

`\correctchoice`, and the other as `\wrongchoice`.
 For this command to work, one has to load package `fp`.
 As an example,

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

produces (in correction mode):

Question 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

```
1911 \def\AMCIntervalFormat#1#2{[#1,\,#2]}
    to follow local conventions (writting  $[a, b[$  instead of  $[a, b[$  is for example a common usage).
1912 \ExplSyntaxOn
1913
1914 \fp_new:N \amc_interv_a
1915 \fp_new:N \amc_interv_b
1916 \cs_new:Npn \amc_intervals:nnnn #1 #2 #3 #4 {
1917   \fp_set:Nn \amc_interv_a { #2 }
1918   \fp_do_while:nn { \amc_interv_a < #3 } {
1919     \fp_set:Nn \amc_interv_b { \amc_interv_a + #4 }
1920     \fp_compare:nTF { \amc_interv_a <= #1 < \amc_interv_b }
1921       \correctchoice \wrongchoice
1922       {\AMCIntervalFormat{\fp_use:N \amc_interv_a}{\fp_use:N \amc_interv_b}}
1923     \fp_set:Nn \amc_interv_a \amc_interv_b
1924   }
1925 }
1926 \cs_new_eq:NN \AMCIntervals \amc_intervals:nnnn
1927
1928 \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:

```

1929 \def\AMCotextGoto{}
1930 \def\AMCotextReserved{}
1931 \def\AMCocol@Background{lightgray}
1932 \def\AMCocol@BoxFrameRule{white}
1933 \def\AMCocol@FrameRule{black}
1934 \def\AMCocol@Foreground{}
1935 \def\AMCopen@answer{}
1936 \def\AMCopen@question{}
1937 \def\AMCopen@lineuptext{}
1938 \define@key{AMCOpen}{backgroundcol}{\def\AMCocol@Background{#1}}
1939 \define@key{AMCOpen}{foregroundcol}{\def\AMCocol@Foreground{#1}}
1940 \define@key{AMCOpen}{Treserved}{\def\AMCotextReserved{#1}}
1941 \define@key{AMCOpen}{question}{\AMCid@name}{\def\AMCopen@question{#1}}
1942 \define@key{AMCOpen}{answer}{\def\AMCopen@answer{#1}}
1943 \define@key{AMCOpen}{contentcommand}[AMCopen@lines]{\def\AMCopen@contentcommand{#1}}
1944 \newdimen\AMCopen@Hspace\AMCopen@Hspace=.5em
1945 \define@key{AMCOpen}{hspace}{\AMCopen@Hspace=#1}
1946 \def\AMCopen@Width{.95\linewidth}
1947 \define@key{AMCOpen}{width}{\def\AMCopen@Width{#1}}
1948 \newdimen\AMCopen@LineHeight\AMCopen@LineHeight=1cm
1949 \define@key{AMCOpen}{lineheight}{\AMCopen@LineHeight=#1}
1950 \newcount\AMCopen@Lines\AMCopen@Lines=1
1951 \define@key{AMCOpen}{lines}{\AMCopen@Lines=#1}
1952 \newdimen\AMCopen@boxmargin\AMCopen@boxmargin=3pt
1953 \define@key{AMCOpen}{boxmargin}{\AMCopen@boxmargin=#1}
1954 \newdimen\AMCopen@boxframerule\AMCopen@boxframerule=1pt
1955 \define@key{AMCOpen}{boxframerule}{\AMCopen@boxframerule=#1}
1956 \define@key{AMCOpen}{boxframerulecol}{\def\AMCocol@BoxFrameRule{#1}}
1957 \define@key{AMCOpen}{framerulecol}{\def\AMCocol@FrameRule{#1}}
1958 \newdimen\AMCopen@framerule\AMCopen@framerule=1pt
1959 \define@key{AMCOpen}{framerule}{\AMCopen@framerule=#1}
1960 \define@key{AMCOpen}{lineuptext}{\def\AMCopen@lineuptext{#1}}

```

```

1961 \define@boolkey{AMCOpen}{dots}[true]{}
1962 \define@boolkey{AMCOpen}{scan}[true]{}
1963 \define@boolkey{AMCOpen}{annotate}[false]{}
1964 \define@boolkey{AMCOpen}{lineup}[false]{}
1965 \setkeys{AMCOpen}{dots,scan,annotate,lineup,contentcommand}
1966 \newcommand\AMCOpenOpts[1]{\setkeys{AMCOpen}{#1}}

```

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

```

1967 \newcommand\AMCopen@lines{%
1968   \begin{minipage}{\AMCopen@Width}%
1969     \loop\vspace{\AMCopen@LineHeight}
1970     \hspace*{.5em}\ifAMC@correc\smash{\AMCopen@answer}\def\AMCopen@answer{}\fi%
1971     \ifKV@AMCopen@dots%
1972     \dotfill\hspace*{.5em}
1973     \fi
1974     \ifnum\AMCopen@Lines>\@ne\par\advance\AMCopen@Lines\m@ne\repeat%
1975   \end{minipage}
1976 }
1977 \newcommand\AMCopenShow[2]{
1978   \ifAMC@ensemble\ifAMCformulaire@dedans%
1979     \AMC@amclog{AUTOQCM[Q=\the\AMCid@quest]^^J}%
1980     \fi\fi%
1981   {\setkeys{AMCOpen}{#1}%
1982     \ifKV@AMCopen@lineup%
1983       \ifAMC@ensemble\else%
1984         \ifx\@empty\AMCopen@lineuptext\@empty\fi%
1985         \fi%
1986         \ifAMC@correc\smash{\AMCopen@answer}\fi%
1987         \ifx\@empty\AMCopen@lineuptext\@empty%
1988           \dotfill%
1989         \else%
1990           \AMCopen@lineuptext\hfill%
1991         \fi%
1992       \else%
1993         \hspace*{.5em}\linebreak[1]\hspace*{\fill}%
1994       \fi%
1995     {\AMCnoCompleteMulti%
1996       \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
1997       \def\AMCanswer##1##2{\ifAMC@ensemble ##1\else%
1998         \ifAMC@inside@box ##1\else{\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
1999       \hspace{\AMCopen@Hspace}}%
2000       \fbboxsep=\AMCopen@boxmargin%
2001       \fbboxrule=\AMCopen@boxframerule%
2002       \fcolorbox{\AMCocol@BoxFrameRule}{\AMCocol@Background}{%
2003         \ifAMC@ensemble\AMCopen@question%
2004         \ifx\@empty\AMCopen@question\@empty\else\hspace{\AMCopen@Hspace}\fi%
2005         \fi%
2006         \begin{choicescustom}[o]%
2007         \ifx\AMCocol@Foreground\@empty\@empty\else%
2008         \def\AMC@boxcolor{\AMCocol@Foreground}%

```

```

2009         \fi%
2010         #2%
2011         \ifKV@AMCOpen@scan\else\AMCdontScan\fi%
2012         \ifKV@AMCOpen@annotate\else\AMCdontAnnotate\fi%
2013     \end{choicescustom}%
2014     \ifx\@empty\AMCotextReserved\@empty%
2015         \hspace{-\AMCopen@Hspace}%
2016     \else%
2017         \ifx\AMCocol@Foreground\@empty\@empty%
2018             \AMCotextReserved%
2019         \else%
2020             \textcolor{\AMCocol@Foreground}{\AMCotextReserved}%
2021         \fi%
2022     \fi%
2023 }%
2024 \ifKV@AMCOpen@lineup\else%
2025     \par\nobreak\noindent%
2026     \hspace*{\fill}{%
2027         \fbboxrule=\AMCopen@framerule%
2028         \fcolorbox{\AMCocol@FrameRule}{white}{%
2029             \csname\AMCopen@contentcommand\endcsname
2030         }%
2031     \vspace{\AMCpostOquest}\par%
2032     \fi%
2033 }%
2034 \ifAMC@ensemble\ifAMCformulaire@dedans%
2035     \AMC@amclog{AUTOQCM[FQ]^^J}%
2036     \fi\fi%
2037 }
2038 \newcommand\AMCopenHide[2]{%
2039     \AMCotextGoto%
2040     \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2041 }
2042 \def\AMCOpen{%
2043     \AMC@if@separate@question{\AMC@mem@category{open}}%
2044     \AMCformatChoices{\AMCopenShow}{\AMCopenHide}%
2045 }

```

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}

```

```

2046 \def\AMCbotextGoto{}
2047 \def\AMCbo@help{}
2048 \define@key{AMCBoxOnly}{help}{\def\AMCbo@help{#1}}
2049 \define@boolkey{AMCBoxOnly}{ordered}[false]{}
2050 \setkeys{AMCBoxOnly}{ordered}
2051 \newcommand\AMCboOpts[1]{\setkeys{AMCBoxOnly}{#1}}
2052 \newcommand\AMCboShow[2]{%
2053   \ifAMC@ensemble\ifAMCformulaire@dedans%
2054     \AMC@amclog{AUTOQCM[Q=\the\AMCid@quest]^^J}%
2055   \fi\fi%
2056   {\setkeys{AMCBoxOnly}{#1}%
2057     \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2058     \def\AMCanswer##1##2{\hspace{\AMCformHSpace} \ifAMC@ensemble ##1\else%
2059       \ifAMC@inside@box ##1\else{\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2060     }%
2061     \ifAMC@ensemble\AMCbo@help\fi%
2062     \ifKV@AMCBoxOnly@ordered%
2063       \begin{choicescustom}[o]%
2064     \else%
2065       \begin{choicescustom}%
2066     \fi%
2067     #2
2068     \end{choicescustom}%
2069   }%
2070   \ifAMC@ensemble\ifAMCformulaire@dedans%
2071     \AMC@amclog{AUTOQCM[FQ]^^J}%
2072   \fi\fi%
2073 }
2074 \newcommand\AMCboHide[2]{
2075   \AMCbotextGoto%
2076   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2077 }
2078 \def\AMCBoxOnly{%
2079   \AMC@if@separate@question{\AMC@mem@category{box}}%
2080   \AMCformatChoices{\AMCboShow}{\AMCboHide}%
2081 }

```

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.

```

2082 \DeclareFontShape{OT1}{cmr}{b}{n}{<35->cmr17}{-}
2083 \def\AMC@watertext{\AMC@loc@draft}
2084 \newcommand\AMCw@termark{%
2085   \setlength{\@tempdimb}{.5\paperwidth}%
2086   \setlength{\@tempdimc}{-.5\paperheight}%
2087   \put(\strip@pt\@tempdimb,\strip@pt\@tempdimc){%
2088     \makebox(0,0){\rotatebox{45}{\AMC@LR{

```



```

2089      \textcolor[gray]{0.8}{
2090      \fontencoding{OT1}\fontfamily{cmr}
2091      \fontseries{b}\fontshape{n}
2092      \fontsize{90pt}{120pt}
2093      \selectfont
2094      \AMC@watertext}}}}}}
2095 \newcommand\AMCw@terprint[1]{%
2096   \setbox\@tempboxa\vbox to \z@{%
2097     \vbox{%
2098       \hbox to \z@{%
2099         #1\hss}}\vss}
2100   \dp\@tempboxa\z@
2101   \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}.

```

2102 \def\AMCcercle#1#2{%
2103   {\setlength{\unitlength}{1mm}%
2104     \begin{picture}(\#1,\#1)(-\#2,-\#2)\thinlines\circle*\#1\end{picture}}}
2105 \def\m@rqueCalage{\AMCcercle{3.6}{1.8}}
2106 \def\m@rque#1{\AMC@tracebox{1}{\#1}{\m@rqueCalage}}
2107 \def\he@dtaille#1{\vbox to 1cm{\#1}}
2108 \def\he@dbas#1{\he@dtaille{\vspace*{\fill}\#1}}
2109 \def\he@dhaut#1{\he@dtaille{\#1\vspace*{\fill}}}
2110 \def\AMC@intituleHead{\AMC@loc@corrected}
2111 \def\AMC@note{}
2112 \def\AMCsetFoot#1{\def\AMC@note{\#1}}
2113 \newcommand\AMCStudentNumber{\the\AMCid@etud}
2114 \newcommand\AMCIDBoxesA{\AMCbin@begin{1}\AMC@binaryBoxes[\AMC@NCBetud]{\the\AMCid@etud}}
2115 \newcommand\AMCIDBoxesB{\AMCbin@begin{2}\AMC@binaryBoxes[\AMC@NCBpage]{\thepage}}
2116 \newcommand\AMCIDBoxesC{\AMCbin@begin{3}\AMC@binaryBoxes[\AMC@NCBcheck]{\the\AMCid@check}}
2117 \newcommand\AMCIDBoxesABC{%
2118   \hbox{\vbox{\noindent\AMCIDBoxesA\
2119     \noindent\AMCIDBoxesB\AMCIDBoxesC}}}%
2120 }
2121 \AtBeginPage{\ifAMC@pagelayout\global\advance\AMCid@check\m@ne%
2122   \ifnum\AMCid@check<1\global\AMCid@check=\AMCid@checkmax\fi%
2123   \AMC@pagepos%
2124   \ifAMC@watermark\ifAMC@correthead\else\AMCw@terprint{\AMCw@termark}%
2125   \fi\fi\fi}
2126 \fancypagestyle{AMCpageHeadOnly}{%
2127   \fancyhf{}\fancyhead[C]{\textsc{\AMC@intituleHead}}}%

```

```

2128 \renewcommand{\headrulewidth}{0pt}%
2129 \renewcommand{\footrulewidth}{0pt}%
2130 }
2131 \fancypagestyle{AMCpageFull}{%
2132 \fancyhf{}%
2133 \fancyhead[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHG}}}}%
2134 \fancyhead[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHD}}}}%
2135 \fancyfoot[L]{\AMC@LR{\leavevmode\m@rque{positionBG}}}%
2136 \fancyfoot[R]{\AMC@LR{\leavevmode\m@rque{positionBD}}}%
2137 \fancyhead[C]{\AMC@LR{\he@dhaut{%
2138 \begin{minipage}[b]{\AMC@CBtaille}\AMCboxColor{black}%
2139 \ifAMCids@top\vbox to \AMCids@height{\texttt{+the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2140 \AMCIDBoxesABC
2141 \end{minipage}%
2142 \ifAMCids@side\hbox to \AMCids@width{\hspace*{\fill}%
2143 \texttt{+the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2144 }}}%
2145 \fancyhfoffset[EOLR]{5mm}%
2146 \fancyfoot[C]{\AMC@note}%
2147 \renewcommand{\headrulewidth}{0pt}%
2148 \renewcommand{\footrulewidth}{0pt}%
2149 }
2150 \newcommand\AMCsubjectPageTag{%
2151 \fbox{\texttt{\the\AMCid@etud:\thepage}}%
2152 }
2153 \fancypagestyle{AMCpageNoMarks}{%
2154 \fancyhf{}%
2155 \fancyhead[R]{\AMCsubjectPageTag}%
2156 \fancyfoot[C]{\AMC@note}%
2157 \renewcommand{\headrulewidth}{0pt}%
2158 \renewcommand{\footrulewidth}{0pt}%
2159 }
2160 \fancypagestyle{AMCpageEmpty}{%
2161 \fancyhf{}%
2162 \renewcommand{\headrulewidth}{0pt}%
2163 \renewcommand{\footrulewidth}{0pt}%
2164 }
2165 \AtBeginDocument{%
2166 \ifAMC@pagelayout%
2167 \ifAMC@correthead
2168 \pagestyle{AMCpageHeadOnly}
2169 \else
2170 \pagestyle{AMCpageFull}
2171 \fi
2172 \fi
2173 }

```

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^AT_EX code $\langle code \rangle$ that generates a single copy can be a little long, so that the environment `examcopy` is often preferred.

```

2174 \newcommand{\onecopy}[2]{%
2175   \ifx\AMCNombreCopies\undefined\AMCnum@copies=#1%
2176   \else\AMCnum@copies=\AMCNombreCopies\fi%
2177   \AMC@amclog{AUTOQCM[TOTAL=\the\AMCnum@copies]^^J}%
2178   \AMCid@etud=\AMCid@etudstart%
2179   \ifnum\AMCid@etud=0\AMCid@etud=\AMC@premierecopie\fi%
2180   \AMCid@etudfin=\AMCnum@copies%
2181   \advance\AMCid@etudfin\AMCid@etud\relax%
2182   \ifAMC@correthead\AMCid@etudfin=\AMC@premierecopie\fi
2183   \ifAMC@pdfform\begin{Form}\fi%
2184   \loop{%
2185     \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
2186       \string\rngstate{\the\AMCid@etud}\the\AMC@SR}%
2187     }\fi%
2188     \AMC@zoneformulairefalse\setcounter{page}{1}\setcounter{section}{0}%
2189     \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageNoMarks}\fi\fi%
2190     \AMCnumero{1}%
2191     \ifAMC@calibration\AMC@amclog{AUTOQCM[ETU=\the\AMCid@etud]^^J}\fi%
2192     \AMC@keepmemoryfalse%
2193     #2%
2194     \ifAMC@keepmemory\else\AMC@mem@clear\fi%
2195     \clearpage}%
2196   \advance\AMCid@etud\@ne\ifnum\AMCid@etud<\AMCid@etudfin\repeat%
2197   \global\AMCid@etudstart=\AMCid@etud%
2198   \ifAMC@pdfform\end{Form}\fi%
2199 }
```

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

```

2200 \newcount\AMC@addpages
2201 \newcommand{\AMCaddpagesto}[1]{%
2202   \AMC@addpages=#1\advance\AMC@addpages\@ne%
2203   \clearpage%
2204   \@whilenum\thepage<\AMC@addpages\do{%
2205     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2206     \hbox{} \clearpage%
2207   }%
2208 }
```

`\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).

```

2209 \def\AMCcleardoublepage{%
```

```

2210 \clearpage%
2211 \ifodd\thepage\else%
2212 \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2213 \hbox{}\clearpage%
2214 \fi%
2215 }

```

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

```

2216 \def\exemplairepair{\ifodd\AMCid@etud}

```

`\AMClabel` Commands `\AMClabel`, `\AMCref` and `\AMCpageref` replaces L^AT_EX's `\label`, `\ref` and `\pageref` to be able to use different labels for different sheets.

```

\AMCref 2217 \newcommand\AMCstudentlabel[1]{\the\AMCid@etud-#1}
2218 \def\AMClabel#1{\expandafter\label{\AMCstudentlabel{#1}}}
2219 \def\AMCref#1{\expandafter\ref{\AMCstudentlabel{#1}}}
2220 \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.

```

2221 \newcommand{\AMCqlabel}[1]{%
2222 \AMClabel{#1}%
2223 }

```

4.18 Pre-association

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

```

2224 \newcommand{\AMCassociation}[2][{}]{%
2225 \ifAMC@calibration%
2226 \immediate\write\AMC@XYFILE{\string\association{\the\AMCid@etud}{#2}{#1}}%
2227 \fi%
2228 }

```

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

```

2229 \newcommand{\AMCstudentslistfile}[2][{}]{%
2230 \ifAMC@calibration%
2231 \immediate\write\AMC@XYFILE{\string\with{studentslistfile=#1}}%
2232 \immediate\write\AMC@XYFILE{\string\with{studentslistkey=#2}}%
2233 \fi%
2234 }

```

4.19 Package options

See section 3.1 for the options descriptions.

```

2235 \def\AMC@lang@code{}

```

```

2236 \DeclareOptionX{noshuffle}{\AMC@ordretrue}
2237 \DeclareOptionX{noshufflegroups}{\AMC@shuffleGfalse}
2238 \DeclareOptionX{fullgroups}{\AMC@fullGroupstrue}
2239 \DeclareOptionX{answers}{\AMC@corretheadtrue\AMC@correcttrue}
2240 \DeclareOptionX{indivanswers}{\AMC@correcttrue}
2241 \DeclareOptionX{textpos}{\AMC@textPostrue}
2242 \DeclareOptionX{extractonly}{\AMC@extractOnlytrue\AMC@textPostrue\AMC@boxedStyle{shape=none}\AMC@boxedAnswers}
2243 \DeclareOptionX{box}{\AMC@qbloctrue}
2244 \DeclareOptionX{asbox}{\AMC@asqbloctrue}
2245 \DeclareOptionX{separateanswersheet}{\AMC@ensembletrue}
2246 \DeclareOptionX{digits}{\AMC@inside@digittrue}
2247 \DeclareOptionX{ordre}{\AMC@ordretrue}
2248 \DeclareOptionX{correc}{\AMC@corretheadtrue\AMC@correcttrue}
2249 \DeclareOptionX{modele}{\AMC@corretheadtrue\AMC@correcfalse\AMC@ordretrue}
2250 \DeclareOptionX{correcindiv}{\AMC@correcttrue}
2251 \DeclareOptionX{init}{\AMC@SR@time}
2252 \DeclareOptionX{bloc}{\AMC@qbloctrue}
2253 \DeclareOptionX{completemulti}{\AMC@complete@multitrue}
2254 \DeclareOptionX{insidebox}{\AMC@inside@boxtrue}
2255 \DeclareOptionX{ensemble}{\AMC@ensembletrue}
2256 \DeclareOptionX{chiffres}{\AMC@inside@digittrue}
2257 \DeclareOptionX{outsidebox}{\AMC@outside@boxtrue}
2258 \DeclareOptionX{calibration}{\AMC@calibrationtrue}
2259 \DeclareOptionX{nowatermark}{\AMC@watermarkfalse}
2260 \newcommand\AMC@catalogMode{%
2261   \AMC@catalogtrue%
2262   \AMC@watermarkfalse\AMC@corretheadtrue%
2263   \AMC@correcttrue\AMC@ordretrue\AMC@shuffleGfalse%
2264   \AMC@fullGroupstrue%
2265   \def\AMC@intituleHead{\AMC@loc@catalog}\AMC@affichekeystrue}
2266 \newcommand\AMC@keys@next{\AMC@keyslinefalse}
2267 \newcommand\AMC@keys@line{\AMC@keyslinetrue}
2268 \DeclareOptionX{catalog}{\AMC@catalogMode}
2269 \DeclareOptionX{keys}[next]{\csname AMC@keys@#1\endcsname{}}
2270 \DeclareOptionX{francais}{\def\AMC@lang@code{FR}\AMC@loc@FR}
2271 \DeclareOptionX{lang}{\def\AMC@lang@code{#1}\csname AMC@loc@#1\endcsname}
2272 \DeclareOptionX{versionA}{-%
2273   \def\AMC@id@checkmax{31}\def\AMC@NCBetud{9}\def\AMC@NCBpage{4}%
2274   \def\AMC@NCBcheck{5}\setlength{\AMC@CBtaille}{4cm}%
2275   \def\AMC@premierecopie{100}}
2276 \DeclareOptionX{plain}{\AMC@plaintrue}
2277 \DeclareOptionX{nopage}{\AMC@pagelayoutfalse}
2278 \DeclareOptionX{postcorrect}{\AMC@postcorrecttrue}
2279 \DeclareOptionX{automarks}{\AMC@automarkstrue}
2280 \newif\ifAMCneeds@storebox\AMCneeds@storeboxfalse
2281 \DeclareOptionX{storebox}{\AMCneeds@storeboxtrue}
2282 \DeclareOptionX{pdfform}{\AMC@pdfformtrue}
2283 \DeclareOptionX{codedigit}{\AMC@codeID@{#1}}
2284 \newif\ifAMC@survey\AMC@surveyfalse
2285 \DeclareOptionX{survey}{\AMC@surveytrue}

```

```

2286 \ProcessOptionsX
2287 \ifAMCneeds@storebox
2288   \RequirePackage{storebox}\AtBeginDocument{{}}%
2289 \fi
2290 \ifAMC@pdfform
2291   \AMC@amclog{AUTOQCM[VAR:project:pdfform=1]^^J}%
2292   \AMC@boxStyle{shape=form}%
2293   \RequirePackage[pageanchor=false]{hyperref}%
2294 \else%
2295   \AMC@amclog{AUTOQCM[VAR:project:pdfform=0]^^J}%
2296 \fi
2297 \AtBeginDocument{%
2298   \ifAMCneeds@storebox%
2299     \let\AMC@new@savebox=\newstorebox%
2300     \let\AMC@save@box=\storebox%
2301     \let\AMC@use@box=\usestorebox%
2302   \fi%
2303   \AMC@new@savebox{\AMC@ovalbox@R}%
2304   \AMC@new@savebox{\AMC@ovalbox@RF}%
2305   \AMC@new@savebox{\AMC@ovalbox@}%
2306   \AMC@new@savebox{\AMC@ovalbox@F}%
2307   \AMC@shapeprepare%
2308 }

```

4.20 Survey add-on

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

Questionnaires

```

auto 2309 \ifAMC@survey
question-auto 2310 \NewEnviron{Questionnaires}[1]{
  values 2311 \onecopy{#1}{
    values-auto 2312 \BODY
  }
variable-auto 2313 }
\answer 2314 }
2315 \NewEnviron{auto}[1]{
2316 \csvreader[head to column names, separator=tab]{#1}{}{
2317 \BODY
2318 }
2319 }
2320 \NewEnviron{question-auto}[3]{
2321 \csvreader[head to column names, separator=tab]{#1}{}{
2322 \ifcsvstrcmp{#2}{#3}{\BODY \\\}{}
2323 }
2324 }
2325 \newenvironment{values}{}{}
2326 \NewEnviron{values-auto}[5]{
2327 \csvreader[head to column names, separator=tab]{#1}{}{
2328 \ifcsvstrcmp{#2}{#3}{

```

```

2329 \ifcsvstrcmp{#4}{#5}{\BODY \
2330 }{}
2331 }{}
2332 }
2333 }
2334 \NewEnviron{variable-auto}[3]{
2335 \foreach \x in {#3}{
2336 \csvreader[head to column names, separator=tab]{#1}{}{
2337 \ifcsvstrcmp{#2}{\x}{\BODY}{}
2338 }
2339 }
2340 }
2341 \newcommand{\answer}[5][\global\advance\AMCrep@count\@ne\relax%
2342 \ifAMC@calibration\AMC@amclog{AUTOQCM[REP=\the\AMCrep@count:B]^^J}\fi%
2343 \global\AMCune@bonnettrue%
2344 \AMCload@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkedbox}%
2345 \else\AMC@box{#1}{}\fi}{#2}{#3}{#4}{#5}}{\the\AMCrep@count}\ignorespaces}
2346 \RequirePackage{tikz}
2347 \usetikzlibrary{positioning, shapes, arrows, tikzmark, decorations.pathreplacing}
2348 \tikzset{
2349     checkbox-sc/.style={
2350         right=of lab\thecsvrow
2351     },
2352     vallab-sc/.style={
2353         text width=4cm,
2354         align=left,
2355     },
2356     checkbox-mc/.style={
2357     },
2358     vallab-mc/.style={
2359         above=of box\thecsvrow,
2360         text width=1.4cm,
2361         align=center,
2362     },
2363     varlab-mc/.style={
2364         text width=4cm,
2365         align=left,
2366     },
2367     node distance= 0mm,
2368 }
2369 \fi

```

4.21 Package Errors

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

```

2370 \def\AMC@error@explain{\PackageError{automultiplechoice}{
2371     Command \protect\explain\space can only be used inside\MessageBreak question like environments}{Something
2372 }}

```

4.22 Optional features

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

```
2373 \ifAMC@plain
2374 \else
2375   \IfFileExists{environ.sty}{\RequirePackage{environ}}{}
2376   \ifx\TeXversion\@undefined
2377   \else
2378     \RequirePackage{etex}
2379   \fi
2380 \fi
```

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

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

```
2381 \@ifpackageloaded{environ}{%
2382   \ifx\NewEnviron\undefined\PackageWarning{automultiplechoice}%
2383   {Package environ loaded but too old version:
2384     environnement examcopy/copieexamen will NOT be defined.}%
2385   \else\NewEnviron{examcopy}[1][5]{\onecopy{#1}{\BODY}}\fi}%
2386 {\PackageWarning{automultiplechoice}%
2387   {Package environ not loaded: environnement
2388     examcopy/copieexamen will NOT be defined.}}
```

4.23 Use with recent LuaTeX versions

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

```
2389 \ExplSyntaxOn
2390
2391 \cs_if_exist:NTF \pdfsavepos { } {
2392   \cs_if_exist:NTF \savepos { \cs_new_eq:NN \pdfsavepos \savepos } { }
2393 }
2394 \cs_if_exist:NTF \pdflastxpos { } {
2395   \cs_if_exist:NTF \lastxpos { \cs_new_eq:NN \pdflastxpos \lastxpos } { }
2396 }
2397 \cs_if_exist:NTF \pdflastypos { } {
2398   \cs_if_exist:NTF \lastypos { \cs_new_eq:NN \pdflastypos \lastypos } { }
2399 }
2400
2401 \ExplSyntaxOff
```


4.24 External control

\SujetExterne \ScoringExterne \CorrigeExterne \CorrigeIndivExterne NoWatermarkExterne	Some of the package options can be controlled defining <code>\xxxExterne</code> commands. For example, the following command will format the subject document, whatever options are used in the L ^A T _E X file: <pre> pdflatex '\nonstopmode\def\SujetExterne{1}\def\NoWatermarkExterne{1}\input{mcq.tex}' 2402 \ifx\SujetExterne\undefined\else 2403 \message{***SUJET***^^J} 2404 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse 2405 \fi 2406 \ifx\ScoringExterne\undefined\else 2407 \message{***SCORING***^^J} 2408 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse\AMC@invisibletrue 2409 \fi 2410 \ifx\CorrigeExterne\undefined\else 2411 \message{***CORRIGE***^^J} 2412 \AMC@calibrationfalse\AMC@corretheadtrue\AMC@correcttrue\AMC@watermarkfalse 2413 \fi 2414 \ifx\CorrigeIndivExterne\undefined\else 2415 \message{***CORRIGE***^^J} 2416 \AMC@calibrationfalse\AMC@corretheadfalse\AMC@correcttrue\AMC@watermarkfalse 2417 \fi 2418 \ifx\CatalogExterne\undefined\else 2419 \message{***CATALOG***^^J} 2420 \AMC@catalogMode 2421 \fi 2422 \ifx\NoWatermarkExterne\undefined\else 2423 \AMC@watermarkfalse 2424 \fi 2425 \ifx\codeDigitExterne\undefined\else 2426 \AMCcodeID@@{\codeDigitExterne} 2427 \fi </pre>
---	--

4.25 Page layout

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

```

2428 \@ifpackageloaded{geometry}{-}{\usepackage{geometry}}
2429 \ifAMC@pagelayout
2430   \ifAMC@correthead
2431     \geometry{hmargin=3cm,vmargin={1cm,1cm},includeheadfoot,headheight=1cm,footskip=1cm}
2432   \else
2433     \geometry{hmargin=3cm,headheight=2cm,headsep=.3cm,footskip=1cm,top=3.5cm,bottom=2.5cm}
2434   \fi
2435 \ifAMC@watermark
2436   \ifAMC@correthead\else
2437     \def\AMC@note{\begin{minipage}{0.65\linewidth}
2438       \AMC@LR{\textcolor{blue}{\AMC@loc@message}}
2439       \end{minipage}}
2440   }

```

```

2441 \fi
2442 \fi
2443 \fi

```

4.26 Initialisation

Initialisation of the check counter:

```
2444 \AMCid@check=\AMCid@checkmax\advance\AMCid@check\@ne
```

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

```

2445 \ifAMC@ensemble\AMC@amclog{AUTOQCM[VAR:ensemble=1]^J}\fi
2446 \ifAMC@inside@box\AMC@amclog{AUTOQCM[VAR:insidebox=1]^J}\fi
2447 \ifAMC@outside@box\AMC@amclog{AUTOQCM[VAR:outsidebox=1]^J}\fi
2448 \ifAMC@postcorrect\AMC@amclog{AUTOQCM[VAR:postcorrect=1]^J}\fi

```

Preparing writing to .xy file :

```

2449 \ifAMC@calibration
2450 \newwrite\AMC@XYFILE%
2451 \immediate\openout\AMC@XYFILE\jobname.xy%
2452 \immediate\write\AMC@XYFILE{\string\version{\AMC@VERSION}}
2453 \immediate\write\AMC@XYFILE{\string\with{codedigit=squarebrackets}}
2454 \immediate\write\AMC@XYFILE{\string\with{version=\AMC@VERSION}}
2455 \immediate\write\AMC@XYFILE{\string\with{ensemble=\ifAMC@ensemble yes\else no\fi}}
2456 \immediate\write\AMC@XYFILE{\string\with{insidebox=\ifAMC@inside@box yes\else no\fi}}
2457 \immediate\write\AMC@XYFILE{\string\with{outsidebox=\ifAMC@outside@box yes\else no\fi}}
2458 \immediate\write\AMC@XYFILE{\string\with{postcorrect=\ifAMC@postcorrect yes\else no\fi}}
2459 \immediate\write\AMC@XYFILE{\string\with{extractonly=\ifAMC@extractOnly yes\else no\fi}}
2460 \immediate\write\AMC@XYFILE{\string\with{lang=\AMC@lang@code}}
2461 \ifx\AMCNombreCopies\undefined%
2462 \immediate\write\AMC@XYFILE{\string\with{ncopies=default}}}%
2463 \else%
2464 \immediate\write\AMC@XYFILE{\string\with{ncopies=\AMCNombreCopies}}}%
2465 \fi%
2466 \fi

```

Preparing writing to .cs file :

```

2467 \ifAMC@catalog%
2468 \newwrite\AMC@CSFILE%
2469 \immediate\openout\AMC@CSFILE\jobname.cs%
2470 \fi%

```

4.27 French command names

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

```

2471 \let\reponses=\choices\let\endreponses=\endchoices
2472 \let\reponseshoriz=\choiceshoriz\let\endreponseshoriz=\endchoiceshoriz
2473 \let\reponsesperso=\choicescustom\let\endreponsesperso=\endchoicescustom
2474 \let\bonne=\correctchoice
2475 \let\mauvaise=\wrongchoice
2476 \let\bareme=\scoring
2477 \let\baremeDefautM=\scoringDefaultM

```

```

2478 \let\baremeDefaultS=\scoringDefaultS
2479 \def\exemplaire{\AMC@loc@FR\onecopy}
2480 \ifpackageloaded{environ}{%
2481   \let\copieexamen=\examcopy\let\endcopieexamen=\endexamcopy}{%
2482   \let\melange groupe=\shuffle group
2483   \let\restitue groupe=\insert group
2484   \let\alafin=\lastchoices
2485   \let\formulaire=\AMCform
2486   \let\AMCdebutFormulaire=\AMCformBegin
2487   \let\champnom=\namefield
2488   \let\choixIntervalles=\AMCIntervals

```

5 Outputs

In the .xy file, 1/⟨*n*⟩ means student sheet number 1 (there is only one “student sheet” for this document as we did not use \onecopy) and page number ⟨*n*⟩ inside this student sheet. Then, each instance of the \tracepos command shows *x* and *y* positions as arguments #2 and #3 (unit is sp, such that 65536×72.27 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{0/35:nom}{0sp}{26831454sp}{square}
\tracepos{0/35:nom}{6038827sp}{0sp}{square}
\tracepos{0/35:nom}{16026323sp}{0sp}{square}
\tracepos{0/35:nom}{0sp}{23846276sp}{square}

```

5.2 AMCboxedchar command

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

```

\tracepos{0/35:test}{22597209sp}{10756930sp}{square}
\tracepos{0/35:test}{23302629sp}{10051510sp}{square}

```

5.3 AMCcode command

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

```

\tracepos{0/59:case:code[5]:16,1}{25579605sp}{40511471sp}{square}
\tracepos{0/59:case:code[5]:16,1}{26285025sp}{39806051sp}{square}
\boxchar{0/59:case:code[5]:16,1}{A}
\tracepos{0/59:case:code[5]:16,2}{25579605sp}{39397359sp}{square}

```

```

\tracepos{0/59:case:code[5]:16,2}{26285025sp}{38691939sp}{square}
\boxchar{0/59:case:code[5]:16,2}{B}
\tracepos{0/59:case:code[5]:16,3}{25579605sp}{38283247sp}{square}
\tracepos{0/59:case:code[5]:16,3}{26285025sp}{37577827sp}{square}
\boxchar{0/59:case:code[5]:16,3}{C}
\tracepos{0/59:case:code[5]:16,4}{25579605sp}{37169135sp}{square}
\tracepos{0/59:case:code[5]:16,4}{26285025sp}{36463715sp}{square}
\boxchar{0/59:case:code[5]:16,4}{D}
\tracepos{0/59:case:code[4]:17,1}{27244404sp}{42739695sp}{square}
\tracepos{0/59:case:code[4]:17,1}{27949824sp}{42034275sp}{square}
\boxchar{0/59:case:code[4]:17,1}{0}
\tracepos{0/59:case:code[4]:17,2}{27244404sp}{41625583sp}{square}
\tracepos{0/59:case:code[4]:17,2}{27949824sp}{40920163sp}{square}
\boxchar{0/59:case:code[4]:17,2}{1}
\tracepos{0/59:case:code[4]:17,3}{27244404sp}{40511471sp}{square}
\tracepos{0/59:case:code[4]:17,3}{27949824sp}{39806051sp}{square}
\boxchar{0/59:case:code[4]:17,3}{2}
\tracepos{0/59:case:code[4]:17,4}{27244404sp}{39397359sp}{square}
\tracepos{0/59:case:code[4]:17,4}{27949824sp}{38691939sp}{square}
\boxchar{0/59:case:code[4]:17,4}{3}
\tracepos{0/59:case:code[4]:17,5}{27244404sp}{38283247sp}{square}
\tracepos{0/59:case:code[4]:17,5}{27949824sp}{37577827sp}{square}
\boxchar{0/59:case:code[4]:17,5}{4}
\tracepos{0/59:case:code[4]:17,6}{27244404sp}{37169135sp}{square}
\tracepos{0/59:case:code[4]:17,6}{27949824sp}{36463715sp}{square}
\boxchar{0/59:case:code[4]:17,6}{5}
\tracepos{0/59:case:code[3]:18,1}{28736261sp}{42739695sp}{square}
\tracepos{0/59:case:code[3]:18,1}{29441681sp}{42034275sp}{square}
\boxchar{0/59:case:code[3]:18,1}{0}
\tracepos{0/59:case:code[3]:18,2}{28736261sp}{41625583sp}{square}
\tracepos{0/59:case:code[3]:18,2}{29441681sp}{40920163sp}{square}
\boxchar{0/59:case:code[3]:18,2}{1}
\tracepos{0/59:case:code[3]:18,3}{28736261sp}{40511471sp}{square}
\tracepos{0/59:case:code[3]:18,3}{29441681sp}{39806051sp}{square}
\boxchar{0/59:case:code[3]:18,3}{2}
\tracepos{0/59:case:code[3]:18,4}{28736261sp}{39397359sp}{square}
\tracepos{0/59:case:code[3]:18,4}{29441681sp}{38691939sp}{square}
\boxchar{0/59:case:code[3]:18,4}{3}
\tracepos{0/59:case:code[3]:18,5}{28736261sp}{38283247sp}{square}
\tracepos{0/59:case:code[3]:18,5}{29441681sp}{37577827sp}{square}
\boxchar{0/59:case:code[3]:18,5}{4}
\tracepos{0/59:case:code[3]:18,6}{28736261sp}{37169135sp}{square}
\tracepos{0/59:case:code[3]:18,6}{29441681sp}{36463715sp}{square}
\boxchar{0/59:case:code[3]:18,6}{5}
\tracepos{0/59:case:code[2]:19,1}{30228118sp}{42739695sp}{square}
\tracepos{0/59:case:code[2]:19,1}{30933538sp}{42034275sp}{square}

```

```

\boxchar{0/59:case:code[2]:19,1}{0}
\tracepos{0/59:case:code[2]:19,2}{30228118sp}{41625583sp}{square}
\tracepos{0/59:case:code[2]:19,2}{30933538sp}{40920163sp}{square}
\boxchar{0/59:case:code[2]:19,2}{1}
\tracepos{0/59:case:code[2]:19,3}{30228118sp}{40511471sp}{square}
\tracepos{0/59:case:code[2]:19,3}{30933538sp}{39806051sp}{square}
\boxchar{0/59:case:code[2]:19,3}{2}
\tracepos{0/59:case:code[2]:19,4}{30228118sp}{39397359sp}{square}
\tracepos{0/59:case:code[2]:19,4}{30933538sp}{38691939sp}{square}
\boxchar{0/59:case:code[2]:19,4}{3}
\tracepos{0/59:case:code[2]:19,5}{30228118sp}{38283247sp}{square}
\tracepos{0/59:case:code[2]:19,5}{30933538sp}{37577827sp}{square}
\boxchar{0/59:case:code[2]:19,5}{4}
\tracepos{0/59:case:code[2]:19,6}{30228118sp}{37169135sp}{square}
\tracepos{0/59:case:code[2]:19,6}{30933538sp}{36463715sp}{square}
\boxchar{0/59:case:code[2]:19,6}{5}
\tracepos{0/59:case:code[1]:20,1}{31719975sp}{42739695sp}{square}
\tracepos{0/59:case:code[1]:20,1}{32425395sp}{42034275sp}{square}
\boxchar{0/59:case:code[1]:20,1}{0}
\tracepos{0/59:case:code[1]:20,2}{31719975sp}{41625583sp}{square}
\tracepos{0/59:case:code[1]:20,2}{32425395sp}{40920163sp}{square}
\boxchar{0/59:case:code[1]:20,2}{1}
\tracepos{0/59:case:code[1]:20,3}{31719975sp}{40511471sp}{square}
\tracepos{0/59:case:code[1]:20,3}{32425395sp}{39806051sp}{square}
\boxchar{0/59:case:code[1]:20,3}{2}
\tracepos{0/59:case:code[1]:20,4}{31719975sp}{39397359sp}{square}
\tracepos{0/59:case:code[1]:20,4}{32425395sp}{38691939sp}{square}
\boxchar{0/59:case:code[1]:20,4}{3}
\tracepos{0/59:case:code[1]:20,5}{31719975sp}{38283247sp}{square}
\tracepos{0/59:case:code[1]:20,5}{32425395sp}{37577827sp}{square}
\boxchar{0/59:case:code[1]:20,5}{4}
\tracepos{0/59:case:code[1]:20,6}{31719975sp}{37169135sp}{square}
\tracepos{0/59:case:code[1]:20,6}{32425395sp}{36463715sp}{square}
\boxchar{0/59:case:code[1]:20,6}{5}

```

Contents

1	Introduction	1
2	Samples	1
2.1	Standard layout	4
2.2	Separate answer sheet	5
2.3	Without markers	6
3	Usage	11
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	22
3.9.3	Answers	22
4	Implementation	22
4.1	Variables	23
4.2	Dimensions	25
4.3	Human readable sheet ID position	26
4.4	Localisation	27
4.4.1	English	27
4.4.2	Catalan	27
4.4.3	Dutch	28
4.4.4	French	28
4.4.5	German	28
4.4.6	Italian	29
4.4.7	Norwegian	29
4.4.8	Portuguese	29
4.4.9	Spanish	30
4.4.10	Japanese	30
4.4.11	Other languages	31
4.5	Interaction with other packages	31
4.5.1	cleveref	31
4.6	Random	31
4.6.1	Random pseudo-generator	31
4.6.2	Uniform random deviates	31
4.6.3	Tokens shuffling	32
4.7	Keys numbering	32
4.8	Boxes	33

4.8.1	Character logging	33
4.8.2	Position logging	33
4.8.3	Boxes to be checked by students	35
4.8.4	Scoring zones	41
4.8.5	Binary boxes	41
4.9	Checking Environment	42
4.10	Handling groups of questions	43
4.11	Questions	47
4.11.1	Managing answers	47
4.11.2	Separate answer sheet	48
4.11.3	Formatting answers	52
4.11.4	Score zones	53
4.11.5	Formatting questions	56
4.11.6	Explanations	58
4.12	Scoring	58
4.13	Numerical data	59
4.13.1	Codes	59
4.13.2	Numerical questions	62
4.13.3	Intervals	75
4.14	Open questions	76
4.15	Boxes with letters only	79
4.16	Page formatting	80
4.16.1	Watermark	80
4.16.2	Signs for scan analysis	81
4.17	Defining a single exam copy content	83
4.18	Pre-association	84
4.19	Package options	84
4.20	Survey add-on	86
4.21	Package Errors	87
4.22	Optional features	88
4.23	Use with recent LuaTeX versions	88
4.24	External control	89
4.25	Page layout	89
4.26	Initialisation	90
4.27	French command names	90
5	Outputs	91
5.1	<code>namefield</code> command	91
5.2	<code>AMCboxedchar</code> command	91
5.3	<code>AMCcode</code> command	91

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		
\"	147, 153	
\@aucune	804, 808, 809	
\@firstoftwo	630	
\@ifstar	968, 969	
\@secondoftwo	632	
\@skiphyperreffalse	1137	
\@skiphyperreftrue	1133, 1137	
\@tempboxa	2096, 2100, 2101	
\@tempdimb	2085, 2087	
\@tempdimc	2086, 2087	
\~	188, 189, 194, 195	
_	1158	
A		
\aa	174	
\alafin	21, 2484	
\amc	831, 842, 844,	
	846, 847, 849, 852, 853,	
	855, 858, 859, 862–865,	
	867, 869, 870, 872, 874,	
	875, 877, 879, 880, 883,	
	884, 886, 887, 889, 890,	
	892, 893, 895, 896, 898,	
	901, 902, 904, 906,	
	910–913, 1215–1219,	
	1221, 1228, 1231, 1236,	
	1240, 1248, 1251–1253,	
	1258, 1260, 1262–	
	1265, 1267, 1272, 1274,	
	1276, 1284, 1289–1291,	
	1293, 1295, 1298–1301,	
	1303, 1305, 1308, 1309,	
	1311, 1312, 1314–1317,	
	1337, 1338, 1343–	
	1348, 1353, 1355–	
	1359, 1361, 1363–	
	1366, 1369, 1376–	
	1378, 1380, 1382, 1383,	
	1385, 1387, 1392, 1393,	
	1395, 1397–1399, 1401,	
	1407, 1412, 1424, 1458,	
	1460, 1462, 1464, 1466,	
	1469, 1472, 1475, 1478,	
	1481, 1484, 1487, 1490,	
	1493, 1502, 1504, 1512,	
	1516, 1518, 1530–	
	1537, 1540, 1543, 1547,	
	1550, 1553, 1557, 1560,	
	1564, 1567, 1572, 1574,	
	1576, 1583, 1585, 1588,	
	1597–1600, 1602, 1604,	
	1607, 1612, 1613, 1615,	
	1619, 1621, 1624–	
	1627, 1629–1631, 1634,	
	1637, 1642, 1644, 1647,	
	1651–1653, 1655, 1658,	
	1663, 1669, 1676, 1677,	
	1684, 1686–1688, 1690,	
	1693, 1695, 1698, 1701,	
	1702, 1707, 1717–	
	1726, 1728, 1733, 1734,	
	1737, 1738, 1742–1744,	
	1746, 1747, 1749–1753,	
	1755, 1756, 1759, 1761,	
	1766–1769, 1771–1777,	
	1780–1783, 1787, 1789,	
	1791, 1794, 1799–1804,	
	1806–1809, 1813, 1814,	
	1816, 1822, 1829, 1834,	
	1836, 1837, 1843, 1852,	
	1854–1857, 1861–1863,	
	1867–1870, 1872–1874,	
	1887, 1889, 1895, 1904,	
	1907, 1908, 1914–	
	1920, 1922, 1923, 1926	
\AMC@addpages	2200, 2202, 2204	
\AMC@affecte	278, 1150	
\AMC@amclog	17,	
	279, 590, 957, 1016,	
	1021, 1168, 1200–1203,	
	1238, 1328, 1520, 1552,	
	1554, 1559, 1561, 1566,	
	1568, 1577, 1590, 1606,	
	1689, 1692, 1708, 1710,	
	1730, 1781, 1788, 1804,	
	1806, 1809, 1811, 1814,	
	1816, 1819, 1823, 1826,	
	1828, 1835, 1840, 1842,	
	1882, 1979, 2035, 2054,	
	2071, 2177, 2191, 2291,	
	2295, 2342, 2445–2448	
\AMC@answerBox	476	
\AMC@answerBox@	357, 484, 531,	
	533, 549, 551, 606, 608	
\AMC@binaryBoxes	600, 2114–2116	
\AMC@box	518, 1018,	
	1019, 1022, 2344, 2345	
\AMC@boxcolor	375, 501, 2008	
\AMC@boxcolor@	375–377, 379,	
	397, 406, 418, 422, 439	
\AMC@boxeddown	389, 489, 500	
\AMC@boxedheight	402, 416, 417,	
	419, 420, 423, 424, 468,	
	491, 495, 496, 508, 511	
\AMC@boxedrule	394, 416–418, 487, 498	
\AMC@boxedwidth	405, 416, 417,	
	419, 420, 423, 424, 468,	
	490, 495, 497, 508, 509	
\AMC@catalogMode	2260, 2268, 2420	
\AMC@CBtaille	598, 2138, 2274	
\AMC@checkbox	357, 378,	
	381, 431, 433, 443, 449,	
	459, 606, 1018, 1521,	
	1541, 1553, 1557, 2344	
\AMC@chiffres	1204	
\AMC@crosschar	400, 503	
\AMC@crossrule	422, 488, 504	
\AMC@CSFILE	287, 2468, 2469	
\AMC@definitnumero	279, 283	
\AMC@draw@crossfalse	380	
\AMC@draw@crosstrue	382	

\AMC@error@explain	\AMC@loc@none 100,	\AMC@new@savebox
. 1195, 1197, <u>2370</u>	113, 126, 140, 154, 167,	. . . 358, 2299, 2303–2306
\AMC@fillcolor@	180, 193, 206, 219, 804	\AMC@note 2111,
. . 378, 396, 397, 413, 418	\AMC@loc@PT 185	2112, 2146, 2156, 2437
\AMC@fin@rep <u>791</u> ,	\AMC@loc@q 96,	\AMC@numeric@alsocorrect
976, 980, 984, 987, 990	109, 123, 136, 150, 164, 1508,
\AMC@formBox <u>518</u>	177, 189, 203, 215, 1140	1825, 1829, 1839, 1843
\AMC@formBox@ <u>518</u>	\AMC@loc@qf 95,	\AMC@numeric@keepas
\AMC@fullGroupsfalse 34	108, 122, 135, 149, 163, 1506, 1818, 1819
\AMC@fullGroupstrue 2238, 2264	176, 188, 202, 214, 813	\AMC@numeric@scoreapprox
\AMC@if@separate@question	\AMC@loc@question . . 101, 1498, 1785, 1795
. 842,	114, 127, 141, 155, 168,	\AMC@numeric@scoreexact
917, 922, 930, 936,	181, 194, 207, 220, 226 1496, 1784, 1790
1322, 1899, 2043, 2079	\AMC@loc@questions . . 102,	\AMC@numeric@scorewrong
\AMC@imax 684, 696–698	115, 128, 142, 156, 169, 1500, 1792, 1795
\AMC@intituleHead	182, 195, 208, 221, 226	\AMC@numeration 278, 282, 283
. 2110, 2127, 2265	\AMC@logchar <u>286</u> , 386	\AMC@outside@sep . . . 493, 499
\AMC@keepmemoryfalse . . 2192	\AMC@logfile 17–19	\AMC@oval@radius
\AMC@keepmemorytrue 958	\AMC@LR <u>21</u> , 388, 418, 492, 509, 511
\AMC@keyBox@	2088, 2133–2137, 2438	\AMC@ovalbox@ . . 432, 446, 2305
. . . 547, 1239, 1523, 1526	\AMC@makeovalbox 411, 430–433	\AMC@ovalbox@F 433, 444, 2306
\AMC@keys@line 2267	\AMC@mem@add 872,	\AMC@ovalbox@R
\AMC@keys@next 2266	918, 925, 932, 939, 1323 429, 430, 452, 2303
\AMC@lang@code	\AMC@mem@add@ifneeded . .	\AMC@ovalbox@RF 431, 450, 2304
. . 2235, 2270, 2271, 2460 <u>829</u> , 1164	\AMC@pagepos <u>292</u> , 2123
\AMC@loc@CA 104	\AMC@mem@addsingle@ifneeded	\AMC@premierecopie
\AMC@loc@catalog 98,	. . 921, 961, 963, 965, 967	. . . 599, 2179, 2182, 2275
111, 125, 138, 152, 166,	\AMC@mem@addvar 877	\AMC@prepare . . . 281, 284, 285
179, 191, 205, 217, 2265	\AMC@mem@aid 896, 931	\AMC@prepare@element . . .
\AMC@loc@corrected . . . 97,	\AMC@mem@answer 928, 647, 655, 765
110, 124, 137, 151, 165,	991, 998–1000, 1005, 1011	\AMC@printformoutside@false
178, 190, 204, 216, 2110	\AMC@mem@category 522
\AMC@loc@DE 145	890, 924, 1899, 2043, 2079	\AMC@printformoutside@true
\AMC@loc@draft 92,	\AMC@mem@clear 847, 2194 524, 525
105, 119, 132, 146, 160,	\AMC@mem@next . . 867, 923, 937	\AMC@printkeyoutside@false
173, 186, 199, 212, 2083	\AMC@mem@openQuestion 539
\AMC@loc@ES 198 935, 1163	\AMC@printkeyoutside@true
\AMC@loc@explain 99, 112,	\AMC@mem@qidaffname . 884, 938 541, 543
139, 153, 192, 218, 1195	\AMC@mem@show . . 911, 948, 957	\AMC@qaff 814, <u>1130</u> , 1158, 1191
\AMC@loc@FR . . 131, 2270, 2479	\AMC@mem@show@filter 912, 952	\AMC@save@box . . 359, 414, 2300
\AMC@loc@IT 159	\AMC@mn@leftmargin	\AMC@setcolors@
\AMC@loc@JA 211	. . 1028, 1036, 1039, 1045 374, 395, 412, 437
\AMC@loc@message 93,	\AMC@mn@rightmargin	\AMC@shape@form 471
106, 120, 133, 147, 161,	. . 1029, 1037, 1040, 1043	\AMC@shape@form@base 458, 472
174, 187, 200, 213, 2438	\AMC@mn@sep . . 1027, 1043, 1045	\AMC@shape@form@ticked . .
\AMC@loc@namesurname . . .	\AMC@mn@test . . 1026, 1033, 1042 460, 462, 465
. . 103, 116, 129, 143,	\AMC@NCBcheck <u>594</u> , 2116, 2274	\AMC@shape@none 475
157, 170, 183, 196, 209	\AMC@NCBetud . <u>594</u> , 2114, 2273	\AMC@shape@oval 436
\AMC@loc@NL 118	\AMC@NCBpage . <u>594</u> , 2115, 2273	\AMC@shape@square 393
\AMC@loc@NO 172		

\AMC@shapename	\AMC@sz@width 1049, 1057, 1062	\AMCbeforeQuestion
.. 292, 300, 309, 318, 494	\AMC@sza@box . 1084, 1092, 1095 <u>1139</u> , 1152, 1223
\AMC@shapename@ 292, 389, 505	\AMC@sza@callin 823, 824,	\AMCbeginAnswer
\AMC@shapeprepare	1106, 1111, 1119, 1121	... 983, <u>1012</u> , 1996, 2057
..... 505, 513, 2307	\AMC@sza@callin@margin 1096	\AMCbeginQuestion
\AMC@shapeprepare@form . 457	\AMC@sza@callin@margins 1100 <u>1139</u> , 1158,
\AMC@shapeprepare@none . 474	\AMC@sza@callin@none .. 1088	1171, 1177, 1191, 1222
\AMC@shapeprepare@oval . 428	\AMC@sza@callin@question 1092	\AMCbin@begin . 609, 2114–2116
\AMC@shapeprepare@square 392	\AMC@sza@callout 819, 820,	\AMCbin@digit
\AMC@shuffletoks	1105, 1110, 1114, 1116	\AMCbin@id . 603, 606, 608, 609
..... 259, 717, 720, 784	\AMC@sza@callout@margin 1095	\AMCbin@endigits
\AMC@smashbox 362, 365–369, 372	\AMC@sza@callout@margins 1099	.. 602, 612, 615, 621, 624
\AMC@smashboxheight	\AMC@sza@callout@none . 1087	\AMCbin@number
..... 363, 366–368, 371	\AMC@sza@callout@question	.. 601, 611, 614, 616, 618
\AMC@smashcentered 1091	\AMCbin@one
.. 364, 406, 407, 439, 440	\AMC@sza@depth 1084, 1104, 1109	\AMCbin@sequence ... 600,
\AMC@SR ... 228, 230, 233–	\AMC@sza@height	611, 616, 617, 622, 623
236, 238, 240, 241, 2186 1084, 1103, 1108	\AMCbin@zero ... 607, 617, 622
\AMC@SR@count 233, 234, 236,	\AMC@sza@init@margin .. 1094	\AMCbloc
244, 248, 251, 254–257	\AMC@sza@init@margins . 1098	\AMCbo@help .. 2047, 2048, 2061
\AMC@SR@time	\AMC@sza@init@none 1086	\AMCboHide
245, 2251	\AMC@sza@init@question 1090	2074, 2080
\AMC@SR@advance . 231, 238, 239	\AMC@sza@width 1084, 1102, 1107	\AMCboOpts
\AMC@SR@bit	\AMC@tempenv	2051
238	628, 629	\AMCboShow
\AMC@SR@const ... 229, 233, 236	\AMC@tmpXY 340, 343, 345	2052, 2080
\AMC@SR@max	\AMC@tracebox	\AMCbotextGoto ... 2046, 2075
<u>244</u> , 272	.. 292, 346, 438, 464,	\AMCboxColor
\AMC@SR@nextByte	582, 584, 587, 1007, 2106	516, 2138
<u>244</u>	\AMC@tracechar . 332, 535, 553	\AMCboxDimensions .. 517, 611
\AMC@SR@num . 246, 247, 249,	\AMC@tracepos .. 293, 403,	\AMCBoxedAnswers .. <u>970</u> , 2242
250, 254, 257, 272, 273	409, 993, 996, 1156, 1164	\AMCBoxOnly
\AMC@SR@set . 230, 242, 243, 245	\AMC@traceposx	<u>2046</u>
\AMC@SR@test	302, 322	\AMCboxOutsideLetter ...
239, 250	\AMC@traceposy . 311, 321, 323 <u>518</u> , 1998, 2059
\AMC@SR@value	\AMC@unnumero	\AMCboxStyle <u>20</u> , <u>487</u> , 2242, 2292
241	<u>278</u>	\AMCcercle
\AMC@stepQuestion	\AMC@use@box	2102, 2105
..... <u>1130</u> , 1149, 1188	360,	\AMCchoiceLabel 476, 557
\AMC@sti 259, 267, 271, 274, 275	444, 446, 450, 452, 2301	\AMCchoiceLabelFormat 390, <u>476</u>
\AMC@stil 260, 268–270, 272, 276	\AMC@VERSION	\AMCcleasedoublepage . <u>17</u> , <u>2209</u>
\AMC@surveyfalse	2452, 2454	\AMCcode
2284	\AMC@watertext ... 2083, 2094	1316
\AMC@surveytrue	\AMC@XYFILE	\AMCcodeGrid
2285	.. 295, 304, 313, 325,	<u>16</u> , <u>1204</u>
\AMC@sz@box	329, 330, 334, 343–345,	\AMCcodeGridInt <u>16</u> , <u>1204</u>
1049,	2185, 2226, 2231, 2232,	\AMCcodeH
1051, 1053, 1056, 1099	2450–2460, 2462, 2464	1317
\AMC@sz@callin	\AMC@XYSpecial	\AMCcodeHspace 1205, 1243, 1278
... 1061, 1066, 1074, 339, 342, 344, 345	\AMCcodeID
1076, 1141, 1159, 1160	\AMCaddpagesto	1210, 1262
\AMC@sz@callin@question 1051	<u>2200</u>	\AMCcodeID@@
\AMC@sz@callout 1060, 1065,	\AMCanswer 1209, 1212, 2283, 2426
1069, 1071, 1153, 1154	.. 1011, <u>1012</u> , 1997, 2058	\AMCcodeID@dot
\AMC@sz@callout@margin 1053	\AMCassociation	1208
\AMC@sz@callout@margins 1056	<u>2224</u>	\AMCcodeID@squarebrackets
\AMC@sz@depth 1049, 1059, 1064	 1207
\AMC@sz@height 1049, 1058, 1063		\AMCcodeVspace
\AMC@sz@init@margins .. 1055		.. 1206, 1241, 1277, 1280

\AMCdebutFormulaire . 21, 2486	\AMCid@etud	\AMClocalized 91
\AMCdecimalPoint	28, 289, 297, 306, 315,	\AMCloop@k 690, 696,
. 1446, 1616, 1648	327, 329, 330, 335, 351,	700, 702, 723, 731, 740
\AMCdefault@groupmode . .	472, 2113, 2114, 2139,	\AMCmarginNote 1030, 1056, 1099
. 644, 658, 659	2143, 2151, 2178, 2179,	\AMCmem@elt@cat 887, 899, 903
\AMCdontAnnotate . . 329, 2012	2181, 2186, 2191, 2196,	\AMCmessage . . . 17, 1151, 1164
\AMCdontScan 329, 2011	2197, 2216, 2217, 2226	\AMCncol@Background 1456, 1851
\AMCdum@reponses . . 784, 798	\AMCid@etudfin	\AMCncol@Border . . 1452, 1851
\AMCemptybox . 1024, 1049, 1084 30, 2180–2182, 2196	\AMCncol@BorderWidth . . .
\AMCendAnswer	\AMCid@etudstart 29, 2178, 2197 1454, 1850
. . . 984, 1012, 1996, 2057	\AMCid@name	\AMCneeds@storeboxfalse 2280
\AMCexponent . 1448, 1859, 1871	. . . 65, 563, 566, 568,	\AMCneeds@storeboxtrue 2281
\AMCform 17, 829, 2485	572, 574, 881, 938, 993,	\AMCnobloc 1145, 1227
\AMCformAfterQuestion . .	996, 1007, 1150, 1164,	\AMCnoCompleteMulti
. 812, 1164	1239, 1523, 1526, 1941 13, 64, 1995
\AMCformAnswer 811	\AMCid@quest	\AMCNombreCopies
\AMCformAnswerA 828, 932	. . . 26, 329, 330, 563,	. . 2175, 2176, 2461, 2464
\AMCformatChoices	566, 568, 572, 574, 582,	\AMCnoScoreZone . . 1129, 1224
. . 1320, 1900, 2044, 2080	584, 587, 880, 938, 993,	\AMCnxtGoto 1437, 1891
\AMCformBeforeQuestion .	996, 1007, 1150, 1151,	\AMCnxtSign 1444, 1667, 1668
. 811, 817	1156, 1164, 1239, 1524,	\AMCnxtVHead . . . 1438, 1620
\AMCformBegin . . 17, 829, 2486	1527, 1730, 1979, 2054	\AMCnum@copies
\AMCformFilter 950	\AMCIDBoxesA 2114, 2118 31, 2175–2177, 2180
\AMCformHSpace . 66, 827, 2058	\AMCIDBoxesABC . . . 2117, 2140	\AMCnum@questions . . 785, 787
\AMCformQuestion 811	\AMCIDBoxesB 2115, 2119	\AMCnumeric@Hspace
\AMCformQuestionA . . 815, 939	\AMCIDBoxesC 2116, 2119	. . . 1439, 1451, 1578,
\AMCformQuestionN . . 814, 822	\AMCids@height . . 76, 87, 2139	1580, 1616, 1635, 1871
\AMCformS 829	\AMCids@sidefalse . . . 79, 81	\AMCnumeric@Vspace . 1440,
\AMCformVSpace 66, 811	\AMCids@sidetrue 83	1450, 1551, 1558, 1565,
\AMCgroup@pre 713, 736	\AMCids@topfalse 79, 83	1592, 1608, 1622,
\AMCgrouploop@next	\AMCids@toptrue 81	1656, 1668, 1858, 1860
. 738, 749, 764	\AMCids@width . . . 75, 86, 2142	\AMCnumericChoices . . 18, 1320
\AMCgrouploop@prep	\AMCidsPosition 73	\AMCnumericChoicesPlain
. 724, 747, 762	\AMCidsVar 77 1898, 1905
\AMCgrouppre@cyclic 707	\AMCidsVarN 77, 78	\AMCnumericHide . . 1895, 1900
\AMCgrouppre@fixed 669	\AMCif@env . . 627, 1195, 1197	\AMCnumericOpts 1516
\AMCgrouppre@withoutreplacement	\AMCifcategory 913	\AMCnumericShow . . 1887, 1900
. 685	\AMCinterBquest 71, 1164	\AMCnumero 1131, 2190
\AMCgrouppre@withreplacement	\AMCinterBrep 66, 997	\AMCocol@Background
. 676	\AMCinterIquest 70, 1164 1931, 1938, 2002
\AMChorizAnswerSep	\AMCinterIrep 66, 974	\AMCocol@BoxFrameRule . .
. 1001, 1002, 1010	\AMCIntervalFormat 1911, 1922 1932, 1956, 2002
\AMChorizBoxSep	\AMCIntervals . 17, 1911, 2488	\AMCocol@Foreground
. . 1003, 1004, 1007, 1009	\AMClabel 2217, 2222 1934, 1939,
\AMCid@check	\AMCload@reponse	2007, 2008, 2017, 2020
. . 27, 327, 2116, 2121,	795, 797, 1018, 1022, 2344	\AMCocol@FrameRule
2122, 2139, 2143, 2444	\AMCload@counter 1933, 1957, 2028
\AMCid@checkmax 25, 773–776, 779,	\AMCOpen 1929
. . . 594, 2122, 2273, 2444	780, 784, 787–790, 793	\AMCopen@answer
	\AMCload@reponse . . . 778, 797	. . 1935, 1942, 1970, 1986

\AMCopen@boxframerule ..	\AMCrep@o	792, 794	B	
..... 1954, 1955, 2001	\AMCrep@perso	982, 991	\bareme	21, 2476
\AMCopen@boxmargin	\AMCrep@r	796	\baremeDefautM	21, 2477
..... 1952, 1953, 2000	\AMCrep@tikz	986, 991	\baremeDefautS	21, 2478
\AMCopen@contentcommand	\AMCrep@tikzmat	989, 991	\bf	1447
..... 1943, 2029	\AMCrien@deux	778, 795	bloc (option)	21
\AMCopen@framerule	\AMCscoreZone	578, 1024	\bonne	21, 2474
..... 1958, 1959, 2027	\AMCscoreZoneAnswerSheet	1024	\bool	905, 1218,
\AMCopen@Hspace ... 1944,	\AMCsection	960	1219, 1240, 1251, 1252,	
1945, 1999, 2004, 2015	\AMCsectionNumbered .	960, 968	1265, 1267, 1276, 1574,	
\AMCopen@LineHeight	\AMCsectionStar	964, 968	1585, 1600, 1619, 1624,	
..... 1948, 1949, 1969	\AMCsetFoot	2112	1665, 1676, 1688, 1706,	
\AMCopen@Lines 1950, 1951, 1974	\AMCsetScoreZone		1707, 1733, 1737, 1746,	
\AMCopen@lines 1082, 1083, 1129		1787, 1822, 1834, 1852	
\AMCopen@lineuptext 1937,	\AMCsetScoreZoneAnswerSheet		box (option)	11
1960, 1984, 1987, 1990 1127–1129		\boxchar	335
\AMCopen@question	\AMCshowSignificantDigits		\boxput	6, 399, 438
.. 1936, 1941, 2003, 2004 1385		C	
\AMCopen@Width 1946, 1947, 1968	\AMCsignificantDigits .	1380	\c	910
\AMCopenHide	\AMCstudentlabel .	2217–2220	calibration (option) ...	5, 11
\AMCopenOpts	\AMCStudentNumber	2113	catalog (option)	11
\AMCopenShow	\AMCstudentslistfile ..	2229	\CatalogExterne	2418
\AMCotextGoto 1929, 2039	\AMCsubjectPageTag 2150, 2155		\champnom	21, 2487
\AMCotextReserved . 1930,	\AMCsubsection	960	\char	1340
1940, 2014, 2018, 2020	\AMCsubsectionNumbered .		\CheckBox	465
\AMCoutsideLabelFormat 962, 969		chiffres (option)	21
518	\AMCsubsectionStar ..	966, 969	\choices	2471
\AMCpageref	\AMCsw@p	259	choices (environment) 13, 970	
\AMCpostOquest	\AMCsw@p@ 261, 263, 265		\choicescustom	2473
72, 2031	\AMCsz@loggedfalse	578	choicescustom (environ-	
\AMCqlabel	\AMCsz@loggedtrue	591	ment)	13, 970
2221	\AMCtext	90	\choiceshoriz	2472
\AMCquestionaff	\AMCtok@ik	690,	choiceshoriz (environment)	
1130	691, 693, 698–702,	 13, 970	
\AMCquestionNumberfalse 1225	722, 730, 743, 750, 766		\choixIntervalles ..	21, 2488
\AMCquestionNumbertrue .. 45	\AMCtok@k 635, 650–652, 656, 766		\cleargroup	15, 756
\AMCrandomseed	\AMCtok@max	636	\clist	1215,
242	\AMCtok@size ... 637, 725–		1228, 1261, 1290, 1299,	
\AMCref	728, 736, 744, 751, 767		1300, 1394, 1397, 1408,	
2217	\AMCw@termark 2082, 2124		1409, 1613, 1614,	
\AMCrep@count . 799, 801, 803	\AMCw@terprint ... 2082, 2124		1644, 1645, 1719, 1720	
\AMCrep@bloc	amcxyfile (environment) .	339	\codeDigitExterne 2425, 2426	
973, 991	\answer	288, 1015, 2309	completemulti (option) ...	11
\AMCrep@count	answers (option)	11	\coordinate	1032
563,	asbox (option)	11	\copieexamen	2481
566, 568, 572, 574, 772,	\association	2226	copieexamen (environment) .	21
781, 801, 803, 807, 893,	auto (environment)	2309	\copygroup	15, 756
931, 972, 978, 982, 986,	automarks (option)	12	\copygroupfrom	15, 756
989, 993, 996, 1007,			correc (option)	21
1015, 1016, 1019–1022,				
1233, 1237–1239, 1321,				
1323, 1519, 1520, 1524,				
1527, 2341, 2342, 2345				
\AMCrep@fini 795, 798, 802, 810				
\AMCrep@init				
791,				
975, 978, 982, 986, 989				
\AMCrep@itemize 972, 991				
\AMCrep@ligne				
978, 991				

correcindiv (option)	21	ensemble (option)	21	1781, 1806, 1914, 1915,
\correctchoice		environments:		1917–1920, 1922, 1923
14, 809, <u>1015</u> , 1921, 2474		amcxyfile	339	francais (option)
\CorrigeExterne	2402	auto	2309	11
\CorrigeIndivExterne . .	2402	choices	13, 970	fullgroups (option)
\crefalias	225	choicescustom . . .	13, 970	12
\crefname	226	choiceshoriz	13, 970	
\cs . . .	842, 846, 847, 849,	copieexamen	21	G
852, 855, 858, 862, 867,		examcopy	2381	\group
869, 872, 874, 877, 879,		question	12, <u>1144</u>	1339, 1351
883, 884, 886, 889, 890,		question-auto	2309	
892, 895, 896, 898, 901,		questionmult	12, <u>1144</u>	H
910–913, 1221, 1231,		Questionnaires	2309	\he@dbas
1236, 1258, 1289, 1293,		questionouverte . . .	<u>1144</u>	2108, 2133, 2134
1298, 1303, 1308, 1311,		reponses	21	\he@dhaut
1314–1317, 1335, 1343,		reponseshoriz	21	2109, 2137
1353, 1356, 1361, 1365,		reponsesperso	21	\he@dtaille
1377, 1380, 1382, 1385,		tikz-multi	970	2107–2109
1387, 1393, 1407, 1412,		tikz-single	970	\hfuzz
1424, 1512, 1516, 1518,		values	2309	368
1532, 1547, 1572, 1583,		values-auto	2309	\ht
1598, 1612, 1642, 1663,		variable-auto	2309	366, 1025
1684, 1728, 1887, 1889,		variable-multi	<u>1144</u>	
1895, 1904, 1907, 1908,		variable-single . . .	<u>1144</u>	I
1916, 1926, 2391, 2392,		\evensidemargin . .	1037, 1039	\ifAMC@affichekeys
2394, 2395, 2397, 2398		\examcopy	2481
\csvreader		examcopy (environment) .	2381	36, 1157, 1158
. .	2316, 2321, 2327, 2336	\exemplaire	21, 2479	\ifAMC@asqbloc . .
		\exemplairepair	2216	42, 812, 818
D		\explain	<u>1193</u> , 2371	\ifAMC@automarks
\define@boolkey		\ExplSyntaxOff		60, 944, 2189, 2205, 2212
. . .	502, 1961–1964, 2049	915, 1319, 1436,	\ifAMC@calibration . . .
\define@choicekey	1897, 1910, 1928, 2401	46,
.	77, 494, 1067, 1112	\ExplSyntaxOn	294, 303, 312, 325, 329,
digits (option)	11	829, 1213, 1333,	330, 333, 1016, 1021,
\ding	466	1441, 1902, 1912, 2389	1151, 1167, 1200–
\dontannotate	330	F		1203, 1238, 2185, 2191,
\dontscan	329	\fancyfoot		2225, 2230, 2342, 2449
\dotfill	354, 1972, 1988	2126, 2131, 2153, 2160	\ifAMC@catalog . .
\draw	418, 422, 1043, 1045	\fbox	2151	47, 385, 2467
		\footrulewidth	\ifAMC@correc 38, 377, 1018,
E		2129, 2148, 2158, 2163	1522, 1970, 1986, 2344
\element	15, <u>635</u>	\foreach	2335	\ifAMC@correchead
\endchoices	2471	\formulaire	21, 2485
\endchoicescustom	2473	\fp	1347, 1355, 1357, 1359,
\endchoiceshoriz	2472	1363, 1368, 1370, 1372,	35, 1194, 2124,
\endcopieexamen	2481	1376, 1383, 1388, 1395,	2167, 2182, 2430, 2436
\endexamcopy	2481	1413, 1416, 1717, 1718,	\ifAMC@draw@cross 361, 400, 421
\endreponses	2471	1742, 1743, 1752, 1780,	\ifAMC@ensemble . .
\endreponseshoriz	2472		54, 523,
\endreponsesperso	2473		540, 561, 580, 832, 944,

<code>\ifAMC@ordre</code>	32, 792	<code>\ifKV@AMCdim@cross</code>	381, 396, 413	<code>\mauvaise</code>	21, 2475
<code>\ifAMC@outside@box</code>	53, 523, 541, 2447, 2457	<code>\ifKV@AMCOpen@annotate</code>	2012	<code>\melangegroupe</code>	21, 2482
<code>\ifAMC@pagelayout</code>	58, 2121, 2166, 2429	<code>\ifKV@AMCOpen@dots</code>	1971	<code>\MessageBreak</code>	2371
<code>\ifAMC@pdfform</code>	62, 349, 2183, 2198, 2290	<code>\ifKV@AMCOpen@lineup</code>	1982, 2024	<code>\multiSymbole</code> <u>1139</u> , 1166, 1184	
<code>\ifAMC@plain</code>	48, 2373	<code>\ifKV@AMCOpen@scan</code>	2011	N	
<code>\ifAMC@postcorrect</code>	59, 809, 2448, 2458	<code>indivanswers</code> (option)	7, 11	<code>\namefield</code>	16, <u>346</u> , 2487
<code>\ifAMC@printformoutside</code>	521, 530	<code>init</code> (option)	11	<code>\namefielddots</code>	<u>347</u>
<code>\ifAMC@printformoutside@</code>	520, 527	<code>\InputIfFileExists</code>	15	<code>\newbox</code>	362
<code>\ifAMC@printkeyoutside</code>	538, 548	<code>\insertgroup</code>	15, <u>716</u> , 2483	<code>\newline</code>	1157
<code>\ifAMC@printkeyoutside@</code>	537, 545	<code>\insertgroupfrom</code>	15, <u>716</u>	<code>\newsavebox</code>	358
<code>\ifAMC@qbloc</code>	41, 1139, 1156, 1164, 1190, 1192, 1892, 2040, 2076	<code>\insidebox</code> (option)	11	<code>\newstorebox</code>	2299
<code>\ifAMC@rbloc</code>	43, 973, 976	<code>\int</code> 844, 846, 850, 856, 863, 902, 1217, 1228, 1262, 1284, 1348, 1364, 1392, 1395, 1397–1399, 1401, 1414, 1417, 1419, 1425, 1426, 1428, 1429, 1431, 1530, 1533, 1535–1537, 1539, 1548, 1573, 1584, 1591, 1597, 1599, 1601, 1607, 1613, 1615, 1620, 1626, 1630, 1634, 1637, 1644, 1646, 1652, 1655, 1658, 1686, 1687, 1690, 1693, 1695, 1698, 1701, 1702, 1721–1723, 1726, 1734, 1737, 1738, 1744, 1749, 1751, 1755, 1768, 1769, 1774, 1782, 1783, 1789, 1791, 1794, 1801, 1808, 1809, 1813, 1814, 1816, 1829, 1836, 1837, 1843, 1857, 1870		<code>\nobreak</code>	519, 2025
<code>\ifAMC@shuffleG</code>	33, 717, 720	K		<code>\node</code>	998, 999, 1180
<code>\ifAMC@survey</code>	2284, 2309	<code>\keys</code>	1247, 1259, 1443, 1513, 1732, 1890	<code>nopage</code> (option)	6, 12
<code>\ifAMC@textPos</code>	39, 993, 996, 1006, 1156, 1164	<code>keys</code> (option)	11	<code>noshuffle</code> (option)	11
<code>\ifAMC@watermark</code>	51, 2124, 2435	L		<code>noshufflegroups</code> (option)	11
<code>\ifAMC@zoneformulaire</code>	57, 525, 562, 833	<code>\lastchoices</code>	<u>791</u> , 2484	<code>\nouveaugroupe</code>	<u>635</u> , 757
<code>\ifAMCcomplete@multi</code>	44, 806	<code>\lastxpos</code>	2395	<code>nowatermark</code> (option)	5, 11
<code>\ifAMCformulaire@dedans</code>	56, 524, 565, 581, 1729, 1881, 1978, 2034, 2053, 2070	<code>\lastypos</code>	2398	<code>\NoWatermarkExterne</code>	<u>2402</u>
<code>\ifAMCids@side</code>	74, 2142	<code>\linebreak</code>	1993	O	
<code>\ifAMCids@top</code>	73, 2139	M		<code>\oddsidemargin</code>	1036, 1040
<code>\ifAMCneeds@storebox</code>	2280, 2287, 2298	<code>\m@rque</code>	2106, 2133–2136	<code>\onecopy</code> <u>2174</u> , 2311, 2385, 2479	
<code>\ifAMCquestionNumber</code>	45, 1137	<code>\m@rqueCalage</code>	2105, 2106	options:	
<code>\ifAMCsz@logged</code>	578, 589	<code>\marginpar</code>	1053, 1095	<code>answers</code>	11
<code>\ifAMCtype@multi</code>	50, 806			<code>asbox</code>	11
<code>\ifAMCune@bonne</code>	49, 808			<code>automarks</code>	12
<code>\ifcase</code>	78			<code>bloc</code>	21
<code>\ifcsname</code>	1068, 1073, 1078, 1113, 1118, 1123			<code>box</code>	11
<code>\ifcsvstrcmp</code>	2322, 2328, 2329, 2337			<code>calibration</code>	5, 11
<code>\ifdim</code>	1042			<code>catalog</code>	11
<code>\ifKV@AMCBoxOnly@ordered</code>	2062			<code>chiffres</code>	21
				<code>completemulti</code>	11
				<code>correc</code>	21
				<code>correcindiv</code>	21
				<code>digits</code>	11
				<code>ensemble</code>	21
				<code>francais</code>	11
				<code>fullgroups</code>	12
				<code>indivanswers</code>	7, 11
				<code>init</code>	11
				<code>insidebox</code>	11
				<code>keys</code>	11
				<code>nopage</code>	6, 12
				<code>noshuffle</code>	11
				<code>noshufflegroups</code>	11
				<code>nowatermark</code>	5, 11
				<code>ordre</code>	21
				<code>outsidebox</code>	11

pdfform	12	reponseshoriz (environ-	503
plain	12	ment)	21
postcorrect	12	\reponsesperso	2473
separateanswersheet	5, 11, 17	reponsesperso (environ-	21
storebox	12	ment)	21
\or	80, 82	\restituegroupe	21, 2483
ordre (option)	21	\rngstate	2186
outsidebox (option)	11		
\ouverte@vs	1144		
		S	
P		\savebox	359
\PackageError	624, 662, 733, 2370	\savepos	2392
\pageref	2220	\sbox	1025
pdfform (option)	12	\scoring	14, 1172, 1178, 1200, 2476
\pgfextractx	1033	\scoringDefaultM	14, 1200, 2477
\pgfpoinanchor	1034	\scoringDefaultS	14, 1200, 2478
\pgfpointdiff	1033	\ScoringExterne	2402
\pgfpointorigin	1033	\section	961, 965
plain (option)	12	separateanswersheet (op-	5, 11, 17
postcorrect (option)	12	tion)	5, 11, 17
\prg	831, 834, 836, 839, 1300, 1396, 1409	\seq	1216, 1264, 1272, 1338, 1346–1348
		\setdefaultgroupmode	658
		\setgroupmode	644, 658
		\shuffle@it	784, 798
Q		\shufflegroup	15, 682, 694, 716, 2482
\question	1144	\shufflegroupslice	699, 702, 719
question (environment)	12, 1144	\smash	1970, 1986
question-auto (environ-	2309	\space	2371
ment)	2309	\storebox	2300
\QuestionIndicative	14, 1200, 1232	storebox (option)	12
questionmult (environment)	12, 1144	\str	899
Questionnaires (environ-	2309	\strut	1043, 1045
ment)	2309	\SujetExterne	2402
questionouverte (environ-	1144		
ment)	1144	T	
		\tex	1341
R		\textasciicircum	1449
\raisebox	1447	\TextField	351
\refstepcounter	1137	\textit	99, 112, 139, 153, 192, 218
\reponses	2471	\textsc	2127
reponses (environment)	21		
\reponseshoriz	2472		
		U	
		\une@rep	972, 973, 978, 982, 986, 989, 1018, 1022, 2344
		\useasboundingbox	416
		\usebox	360
		\usestorebox	2301
		\usetikzlibrary	2347
		V	
		values (environment)	2309
		values-auto (environment)	2309
		variable-auto (environ-	2309
		ment)	2309
		variable-multi (environ-	1144
		ment)	1144
		variable-single (environ-	1144
		ment)	1144
		\version	2452
		\vfuzz	368
		\vrule	1672
		\vtop	1267
		W	
		\wd	368, 369, 1025
		\with	2231, 2232, 2453–2460, 2462, 2464
		\wrongchoice	14, 808, 809, 1015, 1921, 2475
		X	
		\x	2335, 2337