

The automultiplechoice package*

Alexis Bienvenue
paamc@passoire.fr

December 21, 2022

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

*This document corresponds to version revision: `r:8e10ec0f` from AMC 1.5.2+git20221221115238

```

        \end{choiceshoriz}
    \end{question}
}

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

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

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

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

```

```

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

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

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

```

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

```

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

```

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

2.1 Standard layout

A set of 30 students sheets can be produced from the following L^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 7.

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

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

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



+2/1/59+

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

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

2.2 Separate answer sheet

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

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

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

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

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

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

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

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

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

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

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

```
\cleargroup{all}
```

```

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

```

```
\clearpage
```

```
\AMCformBegin
```

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

```
\vspace{2ex}
```

```
\AMCform
```

```

}
\end{document}

```

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

2.3 Without markers

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

The \LaTeX source `sample-plain.tex` that only differs from `sample-amc.tex` by its options passed to `automultiplechoice`:

```
\usepackage[nopage,indivanswers]{automultiplechoice}
```

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

First pages from L^AT_EX source detailed in section 2.1 – see sample-amc.pdf

●  ● +1/1/60+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Egypt?

☐ Cayenne
☐ Caracas
☐ Cuito
☐ Conakry
☐ Chisinau

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

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

Question 4 What is the capital of Ghana?

☐ Accra ☐ Addis Ababa ☐ Ankara ☐ Apia

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

●  ● +2/1/60+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Ireland?

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

Question 3 What is the capital of Ghana?

☐ Apia ☐ Accra ☐ Addis Ababa ☐ Ankara

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

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

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

●  ● +3/1/60+

AMC SAMPLE TEST

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

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

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

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

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

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

Question 3 What is the capital of Ireland?

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

Question 4 What is the capital of Thailand?

☐ Beijing ☐ Bangor ☐ Bangkok ☐ Beirut ☐ Berlin

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

●  ● +4/1/60+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Egypt?

☐ Caracas
☐ Cayenne
☐ Cuito
☐ Conakry
☐ Chisinau

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

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

Question 4 What is the capital of Ireland?

☐ Djibouti ☐ Dhaka ☐ Dakar
☐ Dili ☐ 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

•  • +1/1/60+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Egypt?

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

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

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

Question 4 What is the capital of Ghana?

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

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

•  • +1/2/60+

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

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

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

•  • +2/1/60+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Ireland?

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

Question 3 What is the capital of Ghana?

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

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

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

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

•  • +2/2/60+

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

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

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

First pages from L^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:

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

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

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

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

Question 2 What is the capital of Egypt?

☐ Cayenne
☐ Caracas
☒ Cairo
☐ Cusco
☐ Chisinau

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

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

Question 4 What is the capital of Ghana?

☒ Accra
☐ Addis Ababa
☐ Ankara
☐ Apia

1

AMC

SAMPLE TEST

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

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

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

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

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

Question 2 What is the capital of Ireland?

☐ Djibouti
☐ Dhaka
☒ Dublin
☐ Dili
☐ Doha
☐ Dakar

Question 3 What is the capital of Ghana?

☐ Apia
☒ Accra
☐ Addis Ababa
☐ Ankara

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

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

1

AMC

SAMPLE TEST

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

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

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

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

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

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

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

Question 3 What is the capital of Iceland?

☐ Dhaka
☐ Dili
☐ Doha
☒ Dublin
☐ Djibouti

Question 4 What is the capital of Thailand?

☐ Beijing
☐ Banjul
☒ Bangkok
☐ Beirut
☐ Berlin

1

AMC

SAMPLE TEST

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

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

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

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

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

Question 2 What is the capital of Egypt?

☐ Caracas
☐ Cayenne
☒ Cairo
☐ Cusco
☐ Chisinau

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

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

Question 4 What is the capital of Iceland?

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

Questions environments

Question 1 What is the elevation of Mount Everest?

- ☐ 8,253 m
☐ 8,810 m
☐ 8,848 m

Question 2 ♣ Which contries are in the Americas?

- ☐ Cambodia
☐ Guatemala
☐ Canada
☐ Switzerland

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

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

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

Additional answer “none” for a single question

Question 3 ♣ Which contries are in the Americas?

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

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

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

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

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

The choices environment

Question 4 ♣ What are the possible uses of latex?

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

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

- environment `choiceshoriz` is chosen for short answers:

The choiceshoriz environment

Question 5 How many legs for an insect?

- ☐ 2 ☐ 6 ☐ 8

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

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

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

3.3 Scoring

`\scoring` Scoring strategies can be given in the L^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 `\scoringDefaultM` documentation to write proper commands for your needs. `\scoring{<score>}` can be used inside a `\question` or `\questionmult` environment to describe the scoring strategy for the question, or after a `\correctchoice` or `\wrongchoice` command to describe score associated to a particular choice. `\scoringDefaultM{<score>}` and `\scoringDefaultS{<score>}` define default scoring strategies for multiple and simple questions. `\QuestionIndicative` tags a question that is not taken into account to compute the mark – for example, it can be used for a question about the way students have enjoyed the course.

3.4 Groups of questions

Several commands are available that allows shuffling questions for each question sheet. They handle groups of questions. These groups will usually contain questions, but can be made of any L^AT_EX content.

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

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

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

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

As an example again without questions:

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

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

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

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

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

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

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

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

3.5 Students identification

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

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

The name field

Name and surname:

.....

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

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

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

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

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

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

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

3.6 Separate answer sheet

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

`\AMCform`

`\AMCcleardoublepage`

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

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

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

See section 2.2 for an example.

3.7 Random computation questions

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

Random computation questions

Question 6 How much are 2 plus 8?

☐ 9 ☒ 10 ☐ 16 ☐ -6

```

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

```

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

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

Pick the right interval

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

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

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

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

Numeric choices

Question 8

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

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

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

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

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

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

Available options that can be used in the second argument of the `\AMCnumericChoices` command are the following (`\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).

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

keepas= $\langle name \rangle$ 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 `\AMCnctextGoto` command, as:

```
\def\AMCnctextGoto{\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.

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.

outsideseq 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,outsideseq=.1em,color=black}
```

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

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

3.9.2 Codes

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

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

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

3.9.3 Answers

Environment **choicescustom** will make use of the three commands **\AMCbeginAnswer** (before the first answer), **\AMCendAnswer** (after the last answer) and **\AMCanswer{<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...

Question 10 $2+2=$ $\left(\begin{array}{ccc} \square & \square & \square \end{array} \right)$ $\begin{array}{ccc} 4 & 3 & 2 \end{array}$

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

4 Implementation

This package uses the following other packages:

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

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

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

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

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

4.1 Variables

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

The package defines the following counters:

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

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

`\AMCid@etud` current student sheet number.

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

`\AMCid@check` current page checking number.

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

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

It also defines the following switches:

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

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

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

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

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

`\ifAMC@keysline` 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\ifAMCune@bonne
50 \newif\ifAMCtype@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@multitrue
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` `\AMCformVSpace` is the amount of vertical space between two questions in a separate answer sheet.

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

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

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

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

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

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

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

```

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

```

4.3 Human readable sheet ID position

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

```

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

```

4.4 Localisation

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

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

```

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

```

4.4.1 English

Text indicating draft exams:

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

Message at page bottom when compiled out of AMC gui:

```

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

```

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

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

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

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

Headers for corrected version and catalog:

```

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

```

Localization text for Explanation

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

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

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

Word for 'question', singular and plural forms:

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

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

Default text to write in the students' name box:

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

4.4.2 Catalan

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

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

4.4.3 Dutch

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

```
119 \def\AMC@loc@NL{
120   \def\AMC@loc@draft{Ontwerp}
121   \def\AMC@loc@message{Gebruik bij uw proefwerk bij voorkeur die
122     documenten welke door auto-multiple-choice zijn aangemaakt.}
123   \def\AMC@loc@qf##1{\textbf{Vraag ##1 :}}
124   \def\AMC@loc@q##1##2{\textbf{Vraag ##1} ##2}
125   \def\AMC@loc@corrected{Correctie}
126   \def\AMC@loc@catalog{Catalogus}
127   \def\AMC@loc@none{Geen van de antwoorden is juist.}
128   \def\AMC@loc@question{vraag}
129   \def\AMC@loc@questions{vragen}
130   \def\AMC@loc@namesurname{Achternaam en voornaam:}
131 }
```

4.4.4 French

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

```
132 \def\AMC@loc@FR{
133   \def\AMC@loc@draft{PROJET}
134   \def\AMC@loc@message{Pour votre examen, imprimez de pr\'}ef\'}erence
```

```

135   les documents compil\’es \’a l’aide de auto-multiple-choice.}
136 \def\AMC@loc@qf##1{\textbf{Question ##1 :}}
137 \def\AMC@loc@q##1##2{\textbf{Question ##1} ##2}
138 \def\AMC@loc@corrected{Correction}
139 \def\AMC@loc@catalog{Catalogue}
140 \def\AMC@loc@explain{\textit{\textbf{Explication : }}}
141 \def\AMC@loc@none{Aucune de ces r’eponses n’est correcte.}
142 \def\AMC@loc@question{question}
143 \def\AMC@loc@questions{questions}
144 \def\AMC@loc@namesurname{Nom et pr’enom :}
145 }

```

4.4.5 German

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

```

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

```

4.4.6 Italian

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

```

160 \def\AMC@loc@IT{
161   \def\AMC@loc@draft{BOZZA}
162   \def\AMC@loc@message{Per l’esame, \’e preferibile stampare i documenti
163     a partire da auto-multiple-choice.}
164   \def\AMC@loc@qf##1{\textbf{Domanda ##1:}}
165   \def\AMC@loc@q##1##2{\textbf{Domanda ##1} ##2}
166   \def\AMC@loc@corrected{Correzione}
167   \def\AMC@loc@catalog{Catalogo}
168   \def\AMC@loc@none{Nessuna risposta \’e giusta.}
169   \def\AMC@loc@question{domanda}
170   \def\AMC@loc@questions{domande}
171   \def\AMC@loc@namesurname{Nome e cognome:}
172 }

```

4.4.7 Norwegian

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

```

173 \def\AMC@loc@NO{
174   \def\AMC@loc@draft{UTKAST}
175   \def\AMC@loc@message{Det anbefales {\aa} skrive ut dokumentet

```

```

176 for gjennomgang \direkte fra auto-multiple-choice.}
177 \def\AMC@loc@qf##1{\textbf{Oppgave ##1 :}}
178 \def\AMC@loc@q##1##2{\textbf{Oppgave ##1} ##2}
179 \def\AMC@loc@corrected{Rettet}
180 \def\AMC@loc@catalog{Katalog}
181 \def\AMC@loc@none{Ingen svar er riktige.}
182 \def\AMC@loc@question{oppgave}
183 \def\AMC@loc@questions{oppgave}
184 \def\AMC@loc@namesurname{Etternavn og fornavn:}
185 }

```

4.4.8 Portuguese

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

```

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

```

4.4.9 Spanish

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

```

199 \def\AMC@loc@ES{
200   \def\AMC@loc@draft{BORRADOR}
201   \def\AMC@loc@message{Para revisi\~on, preferentemente imprimir documentos compilados
202     desde auto-multiple-choice.}
203   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
204   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
205   \def\AMC@loc@corrected{Correcci\~on}
206   \def\AMC@loc@catalog{Cat\~alogo}
207   \def\AMC@loc@none{Ninguna de estas preguntas son correctas.}
208   \def\AMC@loc@question{pregunta}
209   \def\AMC@loc@questions{preguntas}
210   \def\AMC@loc@namesurname{Nombre y apellidos:}
211 }

```

4.4.10 Japanese

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

```

212 \def\AMC@loc@JA{
213   \def\AMC@loc@draft{\diamond\diamond\diamond\diamond}
214   \def\AMC@loc@message{\diamond\diamond\diamond\diamondauto-multiple-choice\diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond}
215   \def\AMC@loc@qf##1{\textbf{\diamond##1:}}

```

```

216 \def\AMC@loc@q##1##2{\textbf{\diamond##1} ##2}
217 \def\AMC@loc@corrected{\diamond}
218 \def\AMC@loc@catalog{\diamond\diamond}
219 \def\AMC@loc@explain{\textit{\textbf{\diamond: }}}
220 \def\AMC@loc@none{\diamond}
221 \def\AMC@loc@question{\diamond}
222 \def\AMC@loc@questions{\diamond}
223 }

```

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:

```

224 \AtBeginDocument{\@ifpackageloaded{cleveref}{%
225     \message{AMC/cleveref integration loaded~J}%
226     \crefalias{AMCQuestionaff}{question}%
227     \crefname{question}{\AMC@loc@question}{\AMC@loc@questions}%
228 }{}}%

```

4.6 Random

4.6.1 Random pseudo-generator

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

```

229 \ifx\AMC@SR\undefined\newcount\AMC@SR\fi
230 \providecommand\AMC@SRconst{2097152}
231 \providecommand\AMC@SRset[1]{\global\AMC@SR#1 \ignorespaces}
232 \providecommand\AMC@SRadvance{%
233     \begingroup%
234     \ifnum\AMC@SR<\AMC@SRconst\relax\AMC@SR@count\z@else\AMC@SR@count\@ne\fi%
235     \ifodd\AMC@SR\advance\AMC@SR@count\@ne\fi%
236     \global\divide\AMC@SR\tw@%
237     \ifodd\AMC@SR@count\global\advance\AMC@SR\AMC@SRconst\relax\fi%
238     \endgroup}
239 \providecommand\AMC@SRbit{\AMC@SRadvance\ifodd\AMC@SR1\else0\fi}
240 \providecommand\AMC@SRtest[2]{\AMC@SRadvance%
241     \ifodd\AMC@SR#2\else#1\fi\ignorespaces}
242 \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>}`.

```

243 \AMC@SRset{1515}
244 \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.

```

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

4.6.3 Tokens shuffling

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

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

```

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

```

4.7 Keys numbering

`\AMC@unnumero` This package allocates a unique integer ID to each question key from the questionnaire.
`\AMC@affecte` The counter `\AMC@numerotation` keeps track of the number of keys which already had an

ID. Command `\AMC@definitnumero{n}{key}` allocates ID *n* to the key *key*. Command `\AMC@prepare{key}` looks if an ID had already been associated to *key*, and, if not, makes a new ID allocation for *key*. Command `\AMC@unnumero{key}` returns the ID associated with *key* (creating one if necessary). Command `\AMC@affecte{key}{\cnt}` give to counter `\cnt` the value of the ID associated to *key* (creating one if necessary).

```

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

```

4.8 Boxes

4.8.1 Character logging

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

```

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

```

4.8.2 Position logging

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

```

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

```



```

311 \fi\fi}
312 \def\AMC@traceposy#1#2{%
313 \ifAMC@calibration\ifx\@empty#1\@empty\else%
314 \pdfsavepos\protected@write\AMC@XYFILE{}\string\tracepos%
315 \the\AMCid@etud/\thepage:#2}%
316 {0sp}%
317 {\noexpand\number\pdfastypos 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{}\string\page%
326 \the\AMCid@etud/\thepage/\the\AMCid@check}%
327 {\the\paperwidth}{\the\paperheight}%
328 {\the\pdfpagewidth}{\the\pdfpageheight}}\fi}

```

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

```

\AMCreTick 331 \newcommand{\AMCdontScan}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontscan{\the\AMCid@etud
332 \newcommand{\AMCdontAnnotate}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontannotate{\the\AM
333 \newcommand{\AMCreTick}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\retick{\the\AMCid@etud,\th
334 %

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

```

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

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

```

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

As an example,

```

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

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

produces the following box:

Name:

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

4.8.3 Boxes to be checked by students

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




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

Depending on the required shape for the boxes, the corresponding

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

command is used.

- `\AMC@answerBox@{K}{1}{test}` produce the box K, writing the lines in the `.xy` file shown in section 5.2.

- `\AMC@answerBox@{K}{\AMC@checkbox}{}` produces 
- `\AMC@answerBox@{}{8}{}` produces 
- `\AMC@answerBox@{K}{8}{1}{testb}` produces  with `\AMC@boxStyle{shape=oval,color=red}`

```

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

```

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

```

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

```

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

```

376 \newcommand\AMC@setcolors@[2]{%
377   \def\AMC@boxcolor@{\AMC@boxcolor}%
378   \ifx\empty#1\empty \def\AMC@boxcolor@{black}\fi%
379   \ifAMC@correc\def\AMC@boxcolor@{black}\fi%
380   \def\AMC@fillcolor@{\ifx #2\AMC@checkbox%
381     \AMC@boxcolor@\else white\fi}%
382   \AMC@draw@crossfalse%
383   \ifKV@AMCdim@cross\ifx #2\AMC@checkbox%
384     \AMC@draw@crosstrue\fi\fi%
385 }
386 \newcommand\AMC@answerBox@[4]{%
387   \ifAMC@catalog%
388     \AMC@logchar{#1}{#4}%
389   \fi%
390   \AMC@LR{\hspace{0pt}%
391     \lower\AMC@boxeddown\hbox{\csname AMC@shape@\AMC@shapename@\endcsname%
392       {\AMCchoiceLabelFormat{#1}{#2}{#3}{#4}}}%
393 }
394 \newcommand\AMC@shapeprepare@square{}
395 \newcommand\AMC@shape@square[4]{%
396   \fboxsep=\z@\fboxrule=\AMC@boxedrule%
397   \AMC@setcolors@{#3}{#2}%
398   \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
399   \fcolorbox{\AMC@boxcolor@}{\AMC@fillcolor@}%
400   {%
401     \boxput*(0,0){%

```

```

402     \ifAMC@draw@cross\AMC@crosschar\fi%
403 }{%
404 \vbox to \AMC@boxedheight{%
405     \AMC@tracepos{#3}{#4}%
406     \vfill%
407     \hbox to \AMC@boxedwidth{\hfill%
408         \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
409         \AMC@smashcentered{#2}%
410         \hfill}\vfill}}%
411 \AMC@tracepos{#3}{#4}}%
412 }

```

`\AMC@makeovalbox{<trace>}{<answer>}{<box>}` prepares an oval frame in the L^AT_EX box `<box>`.

```

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

```

```

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

```

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

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

CchoiceLabelFormat 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}}}. .

```

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

```

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

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

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

```

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

```

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

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

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

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

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

```

520 \def\AMCoutsideLabelFormat#1{#1}
521 \newcommand\AMCboxOutsideLetter[2]{#1\nobreak\hspace{.1em}\AMCoutsideLabelFormat{#2}}
522 \newif\ifAMC@printformoutside@%
523 \newcommand\ifAMC@printformoutside{%
524   \AMC@printformoutside@false%
525   \ifAMC@ensemble\ifAMC@outside@box%
526     \ifAMC@formulaire@dedans\AMC@printformoutside@true\fi%
527     \ifAMC@zoneformulaire\AMC@printformoutside@true\fi%

```

```

528 \fi\fi%
529 \ifAMC@printformoutside%
530 }
531 \newcommand\AMC@formBox@[4]{%
532 \ifAMC@printformoutside% letter to be written outside the box
533 \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
534 \else%
535 \AMC@answerBox@{#1}{#2}{#3}{#4}%
536 \fi%
537 \AMC@tracechar{#1}{#2}{#3}{#4}%
538 }
539 \newif\ifAMC@printkeyoutside%
540 \newcommand\ifAMC@printkeyoutside{%
541 \AMC@printkeyoutside@false%
542 \ifAMC@ensemble%
543 \ifAMC@outside@box\AMC@printkeyoutside@true\fi%
544 \else%
545 \ifAMC@inside@box\else\AMC@printkeyoutside@true\fi%
546 \fi%
547 \ifAMC@printkeyoutside%
548 }
549 \newcommand\AMC@keyBox@[4]{%
550 \ifAMC@printkeyoutside%
551 \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
552 \else%
553 \AMC@answerBox@{#1}{#2}{#3}{#4}%
554 \fi%
555 \AMC@tracechar{#1}{#2}{#3}{#4}%
556 }
557 \newcommand\AMC@formBox[4]{%
558 \AMC@formBox@{\ifx\@empty#1\@empty%
559 \AMCchoiceLabel{AMC@ncase}%
560 \else #1\fi}{#2}{#3}{#4}%
561 }
562 \newcommand{\AMC@box}[2]{%
563 \ifAMC@ensemble%
564 \ifAMC@zoneformulaire% for codes inside form sheet
565 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
566 \else%
567 \ifAMC@formulaire@dedans% for answer boxes inside form sheet
568 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
569 \else% outside form sheet: not to be read during data capture
570 \AMC@formBox{#1}{#2}{1}{casequestion:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
571 \fi\fi%
572 \else% no separate sheet for answers: always read
573 \ifAMC@inside@box%
574 \AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
575 \else%
576 \AMC@formBox@{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
577 \fi%
578 \fi%
579 }

```

4.8.4 Scoring zones

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

```

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

4.8.5 Binary boxes

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

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

The length of zone reserved for binary boxes is `\AMC@CBtaille`.

```

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

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

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

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

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

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

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

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



```

602 \newtoks\AMCbin@sequence
603 \newcount\AMCbin@number
604 \newcount\AMCbin@digit
605 \newcount\AMCbin@id
606 \newcount\AMCbin@did
607 \newcount\AMCbin@ndigits
608 \newdimen\AMCbin@hsep
609 \define@key{AMCbin}{ndigits}{\AMCbin@ndigits=#1}
610 \define@key{AMCbin}{id}{\AMCbin@id=#1}
611 \define@key{AMCbin}{hsep}{\AMCbin@hsep=#1}
612 \define@key{AMCbin}{style}[]{\def\AMCbin@style{#1}}
613 \def\AMCbin@one{%
614   \ifnum\AMCbin@did>\z@%
615     \hspace{\AMCbin@hsep}%
616   \fi%
617   \advance\AMCbin@did\@ne%
618   \ifnum\AMCbin@id>0%
619     \AMC@answerBox@{\AMC@checkbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
620   \else%
621     \AMC@answerBox@{\AMC@checkbox}{1}{}%
622   \fi}
623 \def\AMCbin@zero{%
624   \ifnum\AMCbin@did>\z@%
625     \hspace{\AMCbin@hsep}%
626   \fi%
627   \advance\AMCbin@did\@ne%
628   \ifnum\AMCbin@id>0%
629     \AMC@answerBox@{}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
630   \else%
631     \AMC@answerBox@{}{1}{}%
632   \fi}
633 \newcommand{\AMC@binaryCode}[2]{%
634 \setkeys{AMCbin}{ndigits=1,hsep=0pt,style}\setkeys{AMCbin}{#1}%
635 \AMCbin@did=\z@%
636 {\AMCboxDimensions{shape=square,size=.32cm,down=0pt,rule=.2pt,cross=false}\expandafter\AMCboxDimensions%
637 \AMCbin@digit=\z@%
638 \loop%
639 \ifnum\AMCbin@number>\z@%
640 \advance\AMCbin@digit\@ne%
641 \ifodd\AMCbin@number\AMCbin@sequence=\expandafter{\expandafter\AMCbin@one\the\AMCbin@sequence}%
642 \else\AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\fi%
643 \divide\AMCbin@number\tw@%
644 \repeat%
645 \loop\relax%
646 \ifnum\AMCbin@digit<\AMCbin@ndigits\advance\AMCbin@digit\@ne%
647 \AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\repeat%
648 \the\AMCbin@sequence%
649 \ifnum\AMCbin@digit>\AMCbin@ndigits\PackageError{automultiplechoice}{Too low AMC@NCB value (got \the\AMCbin@ndigits)}{}%
650 }}

```

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

```

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

```

4.9 Checking Environment

`\AMCcurrentenv` Sets the current environment as document.

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

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

```

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

```

4.10 Handling groups of questions

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

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

```

664 \newcount\AMCtok@k
665 \newcount\AMCtok@max
666 \newcount\AMCtok@size
667 \newcommand{\nouveaugroupe}[2]{%
668   \expandafter\ifx\csname #1@k\endcsname\relax%
669     \expandafter\newcount\csname #1@k\endcsname%
670     \expandafter\newcount\csname AMC#1@j\endcsname%
671     \csname #1@k\endcsname=\z@\relax%
672     \csname AMC#1@j\endcsname=\z@\relax%
673     \setgroupmode{#1}{\AMCdefault@groupmode}%
674   \fi%
675 }
676 \newcommand\AMC@prepare@element[1]{%
677   \nouveaugroupe{#1}{}%
678   \global\advance\csname #1@k\endcsname\@ne\relax%
679   \AMCtok@k=\csname #1@k\endcsname%
680   \expandafter\ifx\csname #1@romannumeral\AMCtok@k\endcsname\relax%
681     \expandafter\newtoks\csname #1@romannumeral\AMCtok@k\endcsname\fi%
682 }
683 \newcommand{\element}[2]{%
684   \AMC@prepare@element{#1}%

```

```

685 \global\csname #1@romannumeral\AMCtok@k\endcsname={#2}%
686 }

```

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

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

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

```

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

```

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

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

```

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

```

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

```

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

```

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

```

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

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

```

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

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

```

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

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

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

```

745 \newcommand{\shufflegroup}[1]{%
746   \ifAMC@shuffleG{\AMC@shuffletoks{\number\csname #1@k\endcsname}{#1@}}\fi%
747 }
748 \newcommand{\shufflegroupslice}[3]{%
749   \ifAMC@shuffleG{\AMC@shuffletoks[#2]{#3}{#1@}}\fi%
```

```

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

```

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

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

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

See section 3.4 for an illustration for these commands.

```

788 \newcommand{\cleargroup}[1]{%
789   \nouveaugroupe{#1}{}%
790   \csname #1@k\endcsname=\z@\relax%
791   \csname AMC#1@j\endcsname=\z@\relax%

```

```

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

```

4.11 Questions

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

```

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

```

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

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

```

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

```

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

```

819 \def\shuffle@it{\AMC@shuffletoks{\number\AMCload@counter}{reponse@}}
820 \newcount\AMCnum@questions
821 \newcommand\AMCdump@reponses{%
822   \global\AMCnum@questions=\AMCload@counter%
823   \@whilenum\AMCload@counter>0\do{%
824     \the\csname reponse@\romannumeral\AMCload@counter\endcsname%
825     \advance\AMCload@counter\m@ne}}

```

4.11.1 Managing answers

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

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

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

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

```

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

```

4.11.2 Separate answer sheet

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

`\AMCformQuestion` First define commands that will announce questions and answers on the separate answer sheet
`\AMCformAnswer` (these commands can be modified by the user): `\AMCformQuestion{<number>}` is responsible for announcing question, and `\AMCformAnswer{<box>}` is responsible for printing the box to be ticked, given as argument `<box>`.

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

```

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

```

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

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

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

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

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

```

864 \ExplSyntaxOn
865
866 \prg_set_conditional:Nnn \amc_if_separate_question: { p , T } {
867   \ifAMC@ensemble
868     \ifAMC@zoneformulaire
869       \prg_return_false:
870     \else
871       \prg_return_true:
872     \fi
873   \else
874     \prg_return_false:
875   \fi
876 }
877 \cs_new_eq:NN \AMC@if@separate@question \amc_if_separate_question:T
878
879 \int_new:N \amc_memory_elts_count
880
881 \cs_new:Nn \amc_clear_memory: { \int_gzero:N \amc_memory_elts_count }
882 \cs_new_eq:NN \AMC@mem@clear \amc_clear_memory:
883
884 \cs_new:Npn \amc_memory_elt_i:n #1 {
885   amc_memory_elts_ \int_to_alph:n { #1 }
886 }
887 \cs_new:Nn \amc_memory_current_elt: {
888   \amc_memory_elt_i:n \amc_memory_elts_count

```



```

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

```

```

942   } { }
943 }
944 }
945 \cs_new:Nn \amc_use_memory: { \amc_use_memory:n { \c_true_bool } }
946 \cs_new_eq:NN \AMC@mem@show \amc_use_memory:
947 \cs_new_eq:NN \AMC@mem@show@filter \amc_use_memory:n
948 \cs_new_eq:NN \AMC@category \amc_if_category_is_p:n
949
950 \ExplSyntaxOff
951 \newcommand\AMC@mem@add@ifneeded[1]{%
952   \AMC@if@separate@question{%
953     \AMC@mem@add{#1}%
954   }%
955 }
956 \newcommand\AMC@mem@addsingle@ifneeded[2]{%
957   \AMC@if@separate@question{%
958     \AMC@mem@next%
959     \AMC@mem@category{#2}%
960     \AMC@mem@add{#1}%
961   }%
962 }
963 \newcommand\AMC@mem@answer[1]{%
964   \addtocounter{AMC@ncase}{1}%
965   \AMC@if@separate@question{%
966     \AMC@mem@aid{\the\AMC@rep@count}%
967     \AMC@mem@add{\AMCformAnswerA{#1}}%
968   }%
969 }
970 \newcommand\AMC@mem@openQuestion{%
971   \AMC@if@separate@question{%
972     \AMC@mem@next%
973     \AMC@mem@qidaffname{\the\AMC@id@quest}{\arabic{AMC@questionaff}}{\AMC@id@name}%
974     \AMC@mem@add{\AMCformQuestionA}%
975   }%
976 }
977 \def\AMCformBegin{%
978   \AMC@zoneformulairetrue\setcounter{section}{0}%
979   \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageFull}\fi\fi%
980 }
981 \newcommand\AMCform{%
982   \ifAMC@ensemble\AMCformulaire@dedanstrue%
983     \AMC@mem@show%
984   \fi}
985 \newcommand\AMCformFilter[1]{%
986   \ifAMC@ensemble\AMCformulaire@dedanstrue%
987     \AMC@mem@show@filter{#1}%
988   \fi}
989 \newif\ifAMC@keepmemory
990 \newcommand\AMCformS{%
991   \ifAMC@ensemble\AMCformulaire@dedanstrue%
992     \AMCmessage{BR=0}\AMC@mem@show%
993     \global\AMC@keepmemorytrue%
994   \fi}

```

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

```

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

```

4.11.3 Formatting answers

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

`choicescustom` (*env.*) 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` (*env.*)

`tikz-multi` (*env.*)

```

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

```

`\AMCrep@bloc` For each of these styles, a corresponding `\AMCrep@xxx{⟨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@tikzmat`

`\AMCrep@itemize`

`\AMCrep@ligne`

`\AMCrep@perso`

```

1026 \newcommand\AMCrep@bloc[2]{\AMC@mem@answer{#1}%
1027   \par%
1028   \ifAMC@textPos\vbox\bgroup\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\hbox{
1029     \noindent\begin{minipage}{\linewidth}%
1030       \begin{itemize}\item{#1} #2\end{itemize}\end{minipage}%
1031     \ifAMC@textPos\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\egroup\AMC@trac
1032     \vspace{\AMCinterBrep}}
1033 \newcommand\AMCrep@tikz[5]{\AMC@mem@answer{#1}\node[#4] (lab\thecsvrow) at (#3) {#2} node[#5] (box\thecsvrow)

```

```

1034 \newcommand\AMCrep@tikzmat[5]{\AMC@mem@answer{#1}\node[#5] (box\thecsvrow) at (#3) {#1} node[#4] (lab\th
1035 \newcommand\AMCrep@itemize[2]{\AMC@mem@answer{#1}\item{#1} #2}
1036 \newlength\AMChorizAnswerSep
1037 \setlength\AMChorizAnswerSep{3em plus 4em}
1038 \newlength\AMChorizBoxSep
1039 \setlength\AMChorizBoxSep{1em}
1040 \newcommand\AMCrep@ligne[2]{\AMC@mem@answer{#1}%
1041   \ifAMC@textPos%
1042     \mbox{\AMC@tracebox{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}{#1\hspace*\AMChorizBoxS
1043   \else%
1044     \mbox{#1\hspace*\AMChorizBoxSep}{#2}%
1045   \fi\hspace*\AMChorizAnswerSep}}
1046 \newcommand\AMCrep@perso[2]{\AMC@mem@answer{#1}\AMCanswer{#1}{#2}}

```

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

```

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

```

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

```

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

```

4.11.4 Score zones

`\AMCscoreZone` The position of the scores on the annotated answer sheets can be defined in the L^AT_EX 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).

```

1059 \newcommand{\AMCemptybox}[3] {%
1060   \sbox0{\wd0=#1\ht0=#2\dp0=#3\relax\box0}}
1061 \newlength\AMC@mn@test
1062 \newlength\AMC@mn@sep\AMC@mn@sep=4mm
1063 \newlength\AMC@mn@leftmargin
1064 \newlength\AMC@mn@rightmargin
1065 \newcommand\AMCmarginNote[1] {%
1066   \begin{tikzpicture}[remember picture,overlay]%
1067     \coordinate (here) at (0,0);%

```

```

1068 \pgfextractx{\AMC@mn@test}{\pgfpointdiff{\pgfpointorigin}%
1069 {\pgfpointanchor{current page}{center}}}%
1070 \ifodd\thepage%
1071 \AMC@mn@leftmargin=\oddsidemargin%
1072 \AMC@mn@rightmargin=\evensidemargin%
1073 \else
1074 \AMC@mn@leftmargin=\evensidemargin%
1075 \AMC@mn@rightmargin=\oddsidemargin%
1076 \fi
1077 \ifdim\AMC@mn@test < 1cm%
1078 \draw (current page.east |- here)+(-\AMC@mn@rightmargin-1in+\AMC@mn@sep,0pt) node[anchor=text,align=left]{\AMC@mn@score}%
1079 \else%
1080 \draw (current page.west |- here)+(0cm,0pt) node[anchor=text,align=right,text width=\AMC@mn@leftmargin]{\AMC@mn@score}%
1081 \fi%
1082 \end{tikzpicture}%
1083 }

```

Define now different ways to place the score zone:

`none` nowhere

`question` right after the question heading

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

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

```

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

```

Let us now set up options handling.

```

1092 \newlength\AMC@sz@width
1093 \newlength\AMC@sz@height
1094 \newlength\AMC@sz@depth
1095 \def\AMC@sz@callout{}
1096 \def\AMC@sz@callin{}
1097 \define@key{AMCsz}{width}{\AMC@sz@width=#1}
1098 \define@key{AMCsz}{height}{\AMC@sz@height=#1}
1099 \define@key{AMCsz}{depth}{\AMC@sz@depth=#1}
1100 \define@key{AMCsz}{calloutside}{\def\AMC@sz@callout{#1}}
1101 \define@key{AMCsz}{callinside}{\def\AMC@sz@callin{#1}}
1102 \define@choicekey{AMCsz}{position}{none,question,margin,margins}{%
1103 \ifcsname AMC@sz@callout@#1\endcsname%
1104 \def\AMC@sz@callout{AMC@sz@callout@#1}%
1105 \else%
1106 \def\AMC@sz@callout{}%
1107 \fi%
1108 \ifcsname AMC@sz@callin@#1\endcsname%
1109 \def\AMC@sz@callin{AMC@sz@callin@#1}%

```

```

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

```

And do the same for \AMCsetScoreZoneAnswerSheet...

```

1119 \newcommand{\AMC@sza@box}{\AMCemptybox{\AMC@sza@width}{\AMC@sza@height}{\AMC@sza@depth}}
1120 %
1121 \newcommand{\AMC@sza@init@none}{}
1122 \newcommand{\AMC@sza@callout@none}{}
1123 \newcommand{\AMC@sza@callin@none}{}
1124 %
1125 \newcommand{\AMC@sza@init@question}{}
1126 \newcommand{\AMC@sza@callout@question}{}
1127 \newcommand{\AMC@sza@callin@question}{\AMCscoreZone{\AMC@sza@box}}
1128 %
1129 \newcommand{\AMC@sza@init@margin}{}
1130 \newcommand{\AMC@sza@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sza@box}}}
1131 \newcommand{\AMC@sza@callin@margin}{}
1132 %
1133 \newcommand{\AMC@sza@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper ma}}
1134 \newcommand{\AMC@sza@callout@margins}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sza@box}}}
1135 \newcommand{\AMC@sza@callin@margins}{}
1136 %
1137 \newlength\AMC@sza@width
1138 \newlength\AMC@sza@height
1139 \newlength\AMC@sza@depth
1140 \def\AMC@sza@callout{}
1141 \def\AMC@sza@callin{}
1142 \define@key{AMCsz}{width}{\AMC@sza@width=#1}
1143 \define@key{AMCsz}{height}{\AMC@sza@height=#1}
1144 \define@key{AMCsz}{depth}{\AMC@sza@depth=#1}
1145 \define@key{AMCsz}{calloutsideside}{\def\AMC@sza@callout{#1}}
1146 \define@key{AMCsz}{callinsideside}{\def\AMC@sza@callin{#1}}
1147 \define@choicekey{AMCsz}{position}{none,question,margin,margins}{%
1148 \ifcsname AMC@sza@callout@#1\endcsname%
1149 \def\AMC@sza@callout{\AMC@sza@callout@#1}%
1150 \else%
1151 \def\AMC@sza@callout{}}%
1152 \fi%
1153 \ifcsname AMC@sza@callin@#1\endcsname%
1154 \def\AMC@sza@callin{\AMC@sza@callin@#1}%
1155 \else%
1156 \def\AMC@sza@callin{}}%
1157 \fi%
1158 \ifcsname AMC@sza@init@#1\endcsname%
1159 \csname AMC@sza@init@#1\endcsname%
1160 \fi%
1161 }

```

```

1162 \newcommand{\AMCsetScoreZoneAnswerSheet}[1]{\setkeys{AMCsza}{#1}}
1163 \AMCsetScoreZoneAnswerSheet{width=1.5em,height=1.5ex,depth=.5ex,position=none}
1164 \newcommand{\AMCnoScoreZone}{\AMCsetScoreZone{position=none}\AMCsetScoreZoneAnswerSheet{position=none}}

```

4.11.5 Formatting questions

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

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

```

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

```

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

```

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

```

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

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

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

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

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

```

1179 \ifx\question\undefined\else\let\question\undefined\fi
1180 \def\AMCnobloc{\AMC@qblocfalse}
1181 \def\AMCbloc{\AMC@qbloctrue}
1182 \newcommand\AMCstartWithQuestion[1]{%
1183   \global\def\AMCid@name{#1}\AMC@affecte{#1}{\AMCid@quest}%
1184   \ifAMC@calibration%
1185     \AMCmessage{Q=\the\AMCid@quest}%
1186     \immediate\write\AMC@XYFILE{\string\question{\the\AMCid@quest}{\AMCid@name}}%
1187   \fi%

```

```

1188 }
1189 \newcommand\AMCexternalQuestion[2]{%
1190   \AMCstartWithQuestion{#1}%
1191   \ifAMC@calibration%
1192     \AMCmessage{B=MAX=#2}%
1193     \AMCmessage{MULT}%
1194     \AMCmessage{FQ}%
1195   \fi%
1196 }
1197 \newenvironment{question}[2][{}]{%
1198   \def\AMCcurrentenv{question}%
1199   \AMC@stepQuestion%
1200   \AMCstartWithQuestion{#2}%
1201   \AMCbeforeQuestion%
1202   \ifx\@empty\AMC@sz@callout\@empty\else%
1203     \csname\AMC@sz@callout\endcsname%
1204   \fi%
1205   \AMCtype@multifalse\ifAMC@qbloc\ifAMC@textPos\vbox\bgroup\AMC@tracepos{1}{qtext:#2:\the\AMCid@quest,0}%
1206   \ifAMC@affichekeys\index{\texttt{#2}}\ifAMC@keyline[\texttt{#2}]\newline\fi\fi%
1207   \AMCbeginQuestion{\ifAMC@affichekeys\ifAMC@ensemble\AMC@qaff\ \fi\ifAMC@keyline\else[\texttt{#2}]\fi}%
1208   \ifx\@empty\AMC@sz@callin\@empty\else%
1209     \csname\AMC@sz@callin\endcsname%
1210   \fi%
1211   \AMCformulaire@dedansfalse\setcounter{AMC@ncase}{0}%
1212   \AMC@mem@openQuestion%
1213   {\ifAMC@qbloc\end{minipage}\ifAMC@textPos\AMC@tracepos{1}{qtext:\AMCid@name:\the\AMCid@quest,0}\egroup\AMC@tracepos{2}{qtext:#2:\the\AMCid@quest,0}%
1214   \newenvironment{questionmult}[1]{%
1215     \AMC@un@bonnefalse\begin{question}[{\multiSymbole}]{#1}%
1216     \AMCtype@multitrue\ifAMC@calibration%
1217     \AMCmessage{MULT}\fi%
1218   }\end{question}}
1219 \newenvironment{variable-single}[2]{%
1220   {\def\AMCbeginQuestion##1##2{}
1221   \begin{questionmult}{#1}\scoring{v=#2}
1222   \begin{tikz-single}[o]
1223   {\end{tikz-single}
1224   \end{questionmult}}
1225   \newenvironment{variable-multi}[4]{%
1226     {\def\AMCbeginQuestion##1##2{}
1227     \begin{questionmult}{#1}\scoring{v=#4}
1228     \begin{tikz-multi}[o]
1229     \node[#3] (var) at (0,0) {#2};
1230     {\end{tikz-multi}
1231     \end{questionmult}}
1232   \newenvironment{questionmultx}[1]{%
1233     \begin{group}\def\multiSymbole{}\begin{questionmult}{#1}%
1234     {\end{questionmult}\end{group}}
1235   \newdimen\ouverte@vs
1236   \newenvironment{questionouverte}[1][3cm]{%
1237     \AMC@stepQuestion%
1238     \AMCtype@multifalse\ouverte@vs=#1%
1239     \ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1240     \AMCbeginQuestion{\AMC@qaff}{}}%

```



```
1241 {\vspace*{\ouverte@vs}\ifAMC@qbloc\end{minipage}\vspace{3ex}\fi}
```

4.11.6 Explanations

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

```
1242 \newcommand{\explain}[1]{%
1243 \ifAMC@correthead%
1244 \AMCif@env{question}{\par\noindent{\AMC@loc@explain #1}}{\AMC@error@explain}\vspace{1ex}%
1245 \else%
1246 \AMCif@env{question}{}{\AMC@error@explain}%
1247 \fi%
1248 }
1249 \newcommand{\explaincontext}[1]{%
1250 \ifAMC@correc%
1251 #1%
1252 \fi%
1253 }
```

4.12 Scoring

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

`\scoringDefaultS` `\scoring{<score>}` details the scoring strategy for current question or current answer, `\scoringDefaultM` `\scoringDefaultS{<score>}` and `\scoringDefaultM{<score>}` gives default scoring strategy for simple and multiple questions, and `\QuestionIndicative` tells that the current question is not no be taken into account in the global mark.

```
1254 \def\scoring#1{\ifAMC@calibration\AMCmessage{B=#1}\fi}
1255 \def\scoringDefaultS#1{\ifAMC@calibration\AMCmessage{BDS=#1}\fi}
1256 \def\scoringDefaultM#1{\ifAMC@calibration\AMCmessage{BDM=#1}\fi}
1257 \def\QuestionIndicative{\ifAMC@calibration\AMCmessage{INDIC}\fi}
```

4.13 Numerical data

4.13.1 Codes

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

As an example,

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

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

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

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

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

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

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

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

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

```

1258 \newcount\AMC@chiffres
1259 \newdimen\AMCcodeHspace\AMCcodeHspace=.5em
1260 \newdimen\AMCcodeVspace\AMCcodeVspace=.5em
1261 \newcommand\AMCcodeID@squarebrackets[2]{#1[#2]}
1262 \newcommand\AMCcodeID@dot[2]{#1.#2}
1263 \newcommand\AMCcodeID@@[1]{%
1264   \expandafter\def\expandafter\AMCcodeID\expandafter{\csname AMCcodeID@#1\endcsname}%
1265   \def\AMCcodeID@mode{#1}%
1266 }
```

```

1267 \AMCcodeID@@{squarebrackets}
1268 \ExplSyntaxOn
1269
1270 \clist_new:N \amc_code_descr_clist
1271 \seq_new:N \amc_code_digits_seq
1272 \int_new:N \amc_code_digit_n_int
1273 \bool_new:N \amc_code_vertical_bool
1274 \bool_new:N \amc_code_top_bool
1275 \bool_new:N \amc_code_multi_bool
1276 \clist_new:N \amc__multi_clist
1277
1278 \cs_new:Npn \amc_code_init:N #1 {
1279   \def\AMCbeginQuestion##1##2{}
1280   \def\AMCbeforeQuestion{}
1281   \AMCnoScoreZone
1282   \AMCquestionNumberfalse
1283   \setlength{\parindent}{0pt}
1284   \AMCnobloc
1285   \int_set:Nn \amc_code_digit_n_int { \clist_count:N #1 }
1286 }
1287
1288 \cs_new:Nn \amc_code_digit_init: {
1289   \QuestionIndicative
1290   \global\AMCrep@count=\z@
1291 }
1292
1293 \cs_new:Npn \amc_code_digit:n #1 {
1294   \global\advance\AMCrep@count\@ne\relax
1295   \ifAMC@calibration\AMCmessage{ REP = \the\AMCrep@count : M }\fi
1296   \hbox{\AMC@keyBox@{#1}{1}{case : \AMCid@name : \the\AMCid@quest , \the\AMCrep@count}}
1297   \bool_if:NTF \amc_code_vertical_bool {
1298     \vspace{\AMCcodeVspace}
1299   }{
1300     \hspace{\AMCcodeHspace}
1301   }
1302 }
1303
1304 \keys_define:nn { amccode } {
1305   vertical .bool_set:N = \amc_code_vertical_bool,
1306   vertical .initial:n = { true },
1307   vertical .default:n = { true },
1308   v .code:n = { \bool_set_true:N \amc_code_vertical_bool },
1309   h .code:n = { \bool_set_false:N \amc_code_vertical_bool },
1310   top .bool_set:N = \amc_code_top_bool,
1311   top .initial:n = { false },
1312   top .default:n = { true },
1313   multi .bool_set:N = \amc_code_multi_bool,
1314   multi .initial:n = { false },
1315   multi .default:n = { true }
1316 }
1317
1318 \cs_new_nopar:Nn \amc_multi_report: {
1319   \ifAMC@calibration

```

```

1320 \immediate\write\AMC@XYFILE{\string\with{multi=\clist_use:Nn\amc__multi_clist{,}}}}
1321 \fi
1322 }
1323 \cs_new_eq:NN \AMC@multi@report \amc_multi_report:
1324 \int_new:N \amc_multi_count_int
1325 \cs_new_nopar:Nn \amc_multi_clear: {
1326   \int_gzero:N \amc_multi_count_int
1327 }
1328 \cs_new_eq:NN \AMC@multiclear \amc_multi_clear:
1329
1330 \cs_new:Npn \amc_code_generate:nNn #1#2#3 {
1331   { \keys_set:nn { amccode } { #3 }
1332     \bool_if:NTF \amc_code_multi_bool {
1333       \clist_gset:Nn \amc__multi_clist { #1 }
1334     } {}
1335     \bool_if:NTF \amc_code_multi_bool { \int_gincr:N \amc_multi_count_int } {}
1336     \amc_code_init:N #2
1337     \clist_map_inline:Nn #2 { % iterates over 'digits'
1338       \begin{question}{
1339         \AMCcodeID{ #1 \bool_if:NTF
1340           \amc_code_multi_bool
1341           { * \int_use:N \amc_multi_count_int } {} }
1342         { \int_use:N \amc_code_digit_n_int }
1343       }
1344       \amc_code_digit_init:
1345       \seq_set_split:Nnn \amc_code_digits_seq {} { ##1 }
1346       \bool_if:NTF \amc_code_vertical_bool {
1347         \hspace{0pt}
1348         \bool_if:NTF \amc_code_top_bool { \vtop } { \vbox }
1349         \bgroup
1350       }{
1351         \hbox\bgroup
1352       }
1353       \seq_map_inline:Nn \amc_code_digits_seq {
1354         % iterates over available characters for 'digit'
1355         \amc_code_digit:n { ####1 }
1356       }
1357       \bool_if:NTF \amc_code_vertical_bool {
1358         \vspace{-\AMCcodeVspace}\egroup
1359         \hspace{\AMCcodeHspace}
1360       }{
1361         \egroup\vspace{\AMCcodeVspace}
1362         \par
1363       }
1364       \end{question}
1365       \int_decr:N \amc_code_digit_n_int
1366     }
1367   }
1368 }
1369
1370 \cs_new:Npn \amc_code_generate:nnn #1#2#3 {
1371   \clist_set:Nn \amc_code_descr_clist { #2 }
1372   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }

```

```

1373 }
1374 \cs_generate_variant:Nn \amc_code_generate:nnn { xxx }
1375 \newcommand{\AMCcodeGrid}[3][]{
1376   \amc_code_generate:xxx { #2 } { #3 } { #1 }
1377 }
1378
1379 \cs_new:Npn \amc_code_generate_integer:nnn #1#2#3 {
1380   \clist_clear:N \amc_code_descr_clist
1381   \prg_replicate:nn { #2 } { \clist_put_right:Nn \amc_code_descr_clist { 0123456789 } }
1382   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1383 }
1384 \cs_generate_variant:Nn \amc_code_generate_integer:nnn { xxx }
1385 \newcommand{\AMCcodeGridInt}[3][]{
1386   \amc_code_generate_integer:xxx { #2 } { #3 } { #1 }
1387 }
1388
1389 \cs_new:Npn \amc_code_generate_integer_v:nn #1#2 {
1390   \amc_code_generate_integer:nnn { #1 } { #2 } { v }
1391 }
1392 \cs_new:Npn \amc_code_generate_integer_h:nn #1#2 {
1393   \amc_code_generate_integer:nnn { #1 } { #2 } { h }
1394 }
1395 \cs_generate_variant:Nn \amc_code_generate_integer_v:nn { xx }
1396 \cs_generate_variant:Nn \amc_code_generate_integer_h:nn { xx }
1397 \cs_new_eq:NN \AMCcode \amc_code_generate_integer_v:xx
1398 \cs_new_eq:NN \AMCcodeH \amc_code_generate_integer_h:xx
1399
1400 \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 11 What is the value of 7×5 ?																				
<table border="1" style="margin: auto; border-collapse: collapse;"> <tr> <td><input type="checkbox"/>0</td><td><input type="checkbox"/>1</td><td><input type="checkbox"/>2</td><td><input checked="" type="checkbox"/>3</td><td><input type="checkbox"/>4</td><td><input type="checkbox"/>5</td><td><input type="checkbox"/>6</td><td><input type="checkbox"/>7</td><td><input type="checkbox"/>8</td><td><input type="checkbox"/>9</td> </tr> <tr> <td><input type="checkbox"/>0</td><td><input type="checkbox"/>1</td><td><input type="checkbox"/>2</td><td><input type="checkbox"/>3</td><td><input type="checkbox"/>4</td><td><input checked="" type="checkbox"/>5</td><td><input type="checkbox"/>6</td><td><input type="checkbox"/>7</td><td><input type="checkbox"/>8</td><td><input type="checkbox"/>9</td> </tr> </table>	<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
<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` op-

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

```

1401 \newcommand\AMCformatChoices[4]{%
1402   \global\AMCrep@count=\z@%
1403   \AMC@if@separate@question{%
1404     \AMC@mem@add{\global\AMCrep@count=\z@%
1405       #1{#3}{#4}}%
1406   }%
1407   \ifAMC@ensemble%
1408     #2{#3}{#4}%
1409     \AMCmessage{QPART}%
1410   \else%
1411     #1{#3}{#4}%
1412   \fi%
1413 }
```

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

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

```

1414 \ExplSyntaxOn
1415
1416 \cs_generate_variant:Nn \tl_replace_once:Nnn { Nxn }
1417
1418 \tl_new:N \amc_ee_tl
1419 \seq_new:N \amc_ee_seq
```

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

```

1420 \group_begin:
1421 \char_set_catcode_other:N E
1422 \tex_lowercase:D
1423 {
1424   \cs_new:Npn \amc_read_scientific:NNn #1 #2 #3 {
1425     \tl_set:Nn \amc_ee_tl { #3 }
1426     \tl_replace_once:Nxn \amc_ee_tl { E } { e }
1427     \seq_set_split:NnV \amc_ee_seq e \amc_ee_tl
1428     \fp_set:Nn #1 { \seq_item:Nn \amc_ee_seq 1 }
1429     \int_set:Nn #2 { \seq_item:Nn \amc_ee_seq 2 }
1430   }
1431 }
1432 \group_end:
1433
1434 \cs_generate_variant:Nn \amc_read_scientific:NNn { NNf, NNx }
1435
1436 \fp_new:N \amc_fulls_fp
1437 \cs_new:Npn \amc_fp_decompose:NNn #1 #2 #3 {
1438   \fp_set:Nn \amc_fulls_fp { #3 }
```

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

```

1439 \amc_read_scientific:NNx #1 #2
1440 { \fp_to_scientific:N \amc_fulls_fp e 0 }
1441 }
1442 \cs_generate_variant:Nn \amc_fp_decompose:NNn { NNx }
1443
1444 \fp_new:N \amc_num_mantissa_fp
1445 \int_new:N \amc_num_exponent_int
1446 \cs_new:Npn \amc_fp_n_significant_digits:Nnn #1 #2 #3 {
1447   \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_exponent_int
1448   { #2 }
1449   \fp_set:Nn #1
1450   { round(\amc_num_mantissa_fp * 10^((#3)-1)) }
1451   \fp_compare:nTF { abs(#1) >= 10^(#3) }
1452   {
1453     \fp_set:Nn #1 { #1 / 10 }
1454   } { }
1455 }
1456
1457 \fp_new:N \amc_num_nsig_fp
1458 \cs_new:Npn \amc_fp_show_n_significant_digits:nn #1 #2 {
1459   \amc_fp_n_significant_digits:Nnn \amc_num_nsig_fp { #1 } { #2 }
1460 }
1461 \cs_new_eq:NN \AMCsignificantDigits \amc_fp_show_n_significant_digits:nn
1462
1463 \cs_new:Npn \amc_fp_show_significant_digits: {
1464   \fp_use:N \amc_num_nsig_fp
1465 }
1466 \cs_new_eq:NN \AMCshowSignificantDigits \amc_fp_show_significant_digits:
1467
1468 \cs_new:Npn \amc_fp_n_digits:Nnn #1 #2 #3 {
1469   \fp_set:Nn #1
1470   { round((#2) * 10^(#3)) }
1471 }
1472
1473 \int_new:N \amc_todigits_int
1474 \cs_new:Npn \amc_fp_to_digits:Nnnn #1 #2 #3 #4 {
1475   \clist_clear:N #1
1476   \int_set:Nn \amc_todigits_int { \fp_eval:n { abs(round(#2)) } }
1477   \prg_replicate:nn { #3 } {
1478     \clist_put_left:Nx #1 { \int_mod:nn \amc_todigits_int { #4 } }
1479     \int_set:Nn \amc_todigits_int
1480     { \int_div_truncate:nn \amc_todigits_int { #4 } }
1481   }
1482   \int_compare:nNnTF \amc_todigits_int = 0 { } {
1483     \message{^^J!~Error:~number~too~large,
1484       ~some~digits~will~be~discarded^^J}
1485   }
1486 }
1487
1488 \cs_new:Npn \amc_invalid_digits:Nn #1 #2 {
1489   \clist_clear:N #1
1490   \prg_replicate:nn { #2 } { \clist_put_left:Nx #1 { -1 } }
1491 }

```

```

1492
1493 \cs_new:Npn \amc_get_fp_sign:Nn #1 #2 {
1494   \fp_compare:nNnTF #2 < 0 {
1495     \int_set:Nn #1 { -1 }
1496   }{
1497     \fp_compare:nNnTF #2 > 0 {
1498       \int_set:Nn #1 { 1 }
1499     }{
1500       \int_set:Nn #1 { 0 }
1501     }
1502   }
1503 }
1504
1505 \cs_new:Npn \amc_get_int_sign:Nn #1 #2 {
1506   \int_compare:nNnTF #2 < 0 {
1507     \int_set:Nn #1 { -1 }
1508   }{
1509     \int_compare:nNnTF #2 > 0 {
1510       \int_set:Nn #1 { 1 }
1511     }{
1512       \int_set:Nn #1 { 0 }
1513     }
1514   }
1515 }
1516
1517 \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:

1518 \def\AMCncontextGoto{}
1519 \def\AMCncontextVHead#1{\emph{b#1}}
1520 \newdimen\AMCnumeric@Hspace\AMCnumeric@Hspace=.5em
1521 \newdimen\AMCnumeric@Vspace\AMCnumeric@Vspace=1ex
1522 \ExplSyntaxOn
1523
1524 \keys_define:nn { amcnumeric } {
1525   Tsign .code:n = {\def\AMCncontextSign{#1}},
1526   Tsign .initial:n = {},
1527   Tpoint .code:n = {\def\AMCdecimalPoint{#1}},
1528   Tpoint .initial:n = { \raisebox{1ex}{\bf .} },
1529   Texponent .code:n = {\def\AMCexponent{#1}},
1530   Texponent .initial:n = { $\times 10^{\textasciicircum} $ },
1531   vspace .code:n = {\AMCnumeric@Vspace=#1},
1532   hspace .code:n = {\AMCnumeric@Hspace=#1},
1533   bordercol .code:n = {\def\AMCncol@Border{#1}},
1534   bordercol .initial:n = { lightgray },
1535   borderwidth .code:n = {\def\AMCncol@BorderWidth{#1}},
1536   borderwidth .initial:n = { 1mm },
1537   backgroundcol .code:n = {\def\AMCncol@Background{#1}},
1538   backgroundcol .initial:n = { white },
1539   digits .int_set:N = \amc_num_ndigits_int,
1540   digits .initial:n = { 3 },

```



```

1541 decimals .int_set:N = \amc_num_decd_int,
1542 decimals .initial:n = { 0 },
1543 exponent .int_set:N = \amc_num_expo_int,
1544 exponent .initial:n = { 0 },
1545 base .int_set:N = \amc_num_base_int,
1546 base .initial:n = { 10 },
1547 sign .bool_set:N = \amc_num_sign_bool,
1548 sign .initial:n = { true },
1549 sign .default:n = { true },
1550 exposign .bool_set:N = \amc_num_exposign_bool,
1551 exposign .initial:n = { true },
1552 exposign .default:n = { true },
1553 strict .bool_set:N = \amc_num_strict_bool,
1554 strict .initial:n = { false },
1555 strict .default:n = { true },
1556 scoring .bool_set:N = \amc_num_scoring_bool,
1557 scoring .initial:n = { true },
1558 scoring .default:n = { true },
1559 ignoreblank .bool_set:N = \amc_num_ignoreblank_bool,
1560 ignoreblank .initial:n = { false },
1561 ignoreblank .default:n = { true },
1562 vertical .bool_set:N = \amc_num_vertical_bool,
1563 vertical .initial:n = { false },
1564 vertical .default:n = { true },
1565 expovertical .bool_set:N = \amc_num_expovertical_bool,
1566 expovertical .initial:n = { false },
1567 expovertical .default:n = { true },
1568 reverse .bool_set:N = \amc_num_reverse_bool,
1569 reverse .initial:n = { false },
1570 reverse .default:n = { true },
1571 vhead .bool_set:N = \amc_num_vhead_bool,
1572 vhead .initial:n = { false },
1573 vhead .default:n = { true },
1574 nozero .bool_set:N = \amc_num_nozero_bool,
1575 nozero .initial:n = { false },
1576 nozero .default:n = { true },
1577 significant .bool_set:N = \amc_num_significant_bool,
1578 significant .initial:n = { false },
1579 significant .default:n = { true },
1580 scoreexact .code:n = {\def\AMC@numeric@scoreexact{#1}},
1581 scoreexact .initial:n = { 2 },
1582 scoreapprox .code:n = {\def\AMC@numeric@scoreapprox{#1}},
1583 scoreapprox .initial:n = { 1 },
1584 scorewrong .code:n = {\def\AMC@numeric@scorewrong{#1}},
1585 scorewrong .initial:n = { 0 },
1586 exact .int_set:N = \amc_num_exact_int,
1587 exact .initial:n = { 0 },
1588 approx .int_set:N = \amc_num_approx_int,
1589 approx .initial:n = { 0 },
1590 keepas .code:n = {\def\AMC@numeric@keepas{#1}},
1591 keepas .initial:n = {},
1592 alsocorrect .code:n = {\def\AMC@numeric@alsocorrect{#1}},
1593 alsocorrect .initial:n = {}

```

```

1594 }
1595
1596 \cs_new:Npn \amc_num_setopts #1 {
1597   \keys_set:nn { amcnumeric } { #1 }
1598 }
1599
1600 \cs_new_nopar:Nn \amc_num_check_score_opts: {
1601   \bool_if:NTF \amc_num_ignoreblank_bool {
1602     \int_compare:nNnTF \amc_num_base_int = { 10 } { } {
1603       \message{^^J!~Error:~ignoreblank~can~only~be~used~with~number~base~10^^J}
1604     }
1605   } {}
1606 }
1607
1608 \cs_new_eq:NN \AMCnumericOpts \amc_num_setopts
1609

```

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

```

1610 \cs_new:Npn \amc_num_char:nn #1 #2 {
1611   \global\advance\AMCrep@count\@ne\relax
1612   \AMCmessage{REP= \the\AMCrep@count :
1613     \ifx#2\AMC@checkedbox B\else M\fi }
1614   \ifAMC@correc
1615     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :
1616       \the\AMCid@quest , \the\AMCrep@count}
1617   \else
1618     \protect\AMC@keyBox@{#1}{}{1}{case : \AMCid@name :
1619       \the\AMCid@quest , \the\AMCrep@count}
1620   \fi
1621 }

```

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...)

```

1622 \int_new:N \amc_num_digit_value_int
1623 \tl_new:N \amc_num_digit_value_tl
1624 \cs_new:Npn \amc_num_digit_box:nn #1 #2 {
1625   \int_set:Nn \amc_num_digit_value_int { #1 }
1626   \tl_set:Nn \amc_num_digit_value_tl {
1627     \int_compare:nNnTF { \amc_num_digit_value_int } < { 10 }
1628     { \int_to_arabic:n { \amc_num_digit_value_int } }
1629     { \int_to_Alph:n { \amc_num_digit_value_int - 9 } }
1630   }
1631   \int_compare:nNnTF { #1 } = { #2 } {
1632     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1633       {\AMC@checkedbox}
1634   } {
1635     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1636       {}
1637   }
1638 }

```

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>`).

```

1639 \cs_new:Npn \amc_num_sign_boxes:Nn #1 #2 {
1640   \int_case:nn { #1 } {
1641     { -1 } {
1642       \hbox{\amc_num_char:nn{+}{}}
1643       \vspace{\AMCnumeric@Vspace}
1644       \AMCmessage{B=set. sign #2 =1}
1645       \hbox{\amc_num_char:nn{-}{\AMC@checkbox}}
1646       \AMCmessage{B=set. sign #2 =-1}
1647     }
1648     { 1 } {
1649       \hbox{\amc_num_char:nn{+}{\AMC@checkbox}}
1650       \vspace{\AMCnumeric@Vspace}
1651       \AMCmessage{B=set. sign #2 =1}
1652       \hbox{\amc_num_char:nn{-}{}}
1653       \AMCmessage{B=set. sign #2 =-1}
1654     }
1655     { 0 } {
1656       \hbox{\amc_num_char:nn{+}{}}
1657       \vspace{\AMCnumeric@Vspace}
1658       \AMCmessage{B=set. sign #2 =1}
1659       \hbox{\amc_num_char:nn{-}{}}
1660       \AMCmessage{B=set. sign #2 =-1}
1661     }
1662   }
1663 }

```

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.

```

1664 \cs_new:Npn \amc_num_digit_boxes_h:nnn #1 #2 #3 {
1665   \int_step_inline:nnnn
1666   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1667   { 1 } { #3 - 1 } {
1668     \amc_num_digit_box:nn { ##1 }{ #2 }
1669     \AMCmessage{B= set. #1 = ##1}
1670     \hspace{\AMCnumeric@Hspace}
1671   }
1672   \hspace{-\AMCnumeric@Hspace}
1673 }
1674
1675 \cs_new:Npn \amc_num_digit_boxes_v:nnn #1 #2 #3 {
1676   \int_step_inline:nnnn
1677   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1678   { 1 } { #3 - 1 } {
1679     \vbox{\hbox{
1680       \amc_num_digit_box:nn { ##1 }{ #2 }
1681     }}
1682     \AMCmessage{B= set. #1 = ##1}
1683     \int_compare:nNnTF { ##1 } < { #3 - 1 } {
1684       \vspace{\AMCnumeric@Vspace}

```

```

1685   } {}
1686 }
1687 }
1688
1689 \int_new:N \amc_num_first_digit_int
1690 \cs_new:Npn \amc_num_digit_boxes_vr:nnn #1 #2 #3 {
1691   \int_set:Nn \amc_num_first_digit_int
1692   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1693   \int_step_inline:nnnn { #3 - 1 } { -1 }
1694   \amc_num_first_digit_int {
1695     \vbox{\hbox{
1696       \amc_num_digit_box:nn { ##1 }{ #2 }
1697     }}
1698     \AMCmessage{B= set. #1 = ##1}
1699     \int_compare:nNnTF { ##1 } > \amc_num_first_digit_int {
1700       \vspace{\AMCnumeric@Vspace}
1701     } {}
1702   }
1703 }

```

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

```

1704 \cs_new:Npn \amc_num_integer_boxes_v:Nnn #1 #2 #3 {
begin a loop over all digits,
1705   \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1706   \clist_map_inline:Nn #1 {
place the decimal point if necessary,
1707     \int_compare:nNnTF \amc_num_digit_int = { #3 } {
1708       \hbox{ \AMCdecimalPoint }\hspace{\AMCnumeric@Hspace}
1709     } { }

```

draw the box for this digit,

```

1710   \hbox{\vbox{
1711     \bool_if:NTF \amc_num_vhead_bool {
1712       \vbox{\hbox{\AMCtextVHead{ \int_eval:n
1713         { \amc_num_digit_int - 1 } }}}
1714       \vspace{\AMCnumeric@Vspace}
1715     } { }
1716     \bool_if:NTF \amc_num_reverse_bool {
1717       \amc_num_digit_boxes_vr:nnn { #2
1718         \int_to_Alph:n \amc_num_digit_int }
1719       { ##1 } { \amc_num_base_int }
1720     } {
1721       \amc_num_digit_boxes_v:nnn { #2
1722         \int_to_Alph:n \amc_num_digit_int }
1723       { ##1 } { \amc_num_base_int }
1724     }
1725   }}

```

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

```

1726   \int_compare:nNnTF \amc_num_digit_int > 1 {
1727     \hspace{\AMCnumeric@Hspace}
1728   } { }

```

```

1729 \int_decr:N \amc_num_digit_int
1730 }
1731 }
1732

```

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

```

1733
1734 \cs_new:Npn \amc_num_integer_boxes_h:Nnn #1 #2 #3 {
1735   \vbox{
1736     \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1737     \clist_map_inline:Nn #1 {
1738       \int_compare:nNnTF
1739         \amc_num_digit_int = { #3 } {
1740           \hbox{ \AMCdecimalPoint }
1741         } { }
1742       \hbox{
1743         \amc_num_digit_boxes_h:nnn { #2
1744           \int_to_Alph:n \amc_num_digit_int }
1745         { ##1 } \amc_num_base_int
1746       }
1747       \int_compare:nNnTF \amc_num_digit_int > 1 {
1748         \vspace{\AMCnumeric@Vspace}
1749       } { }
1750       \int_decr:N \amc_num_digit_int
1751     }
1752   }
1753

```

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

```

1754
1755 \cs_new:Npn \amc_num_integer_boxes:NnnNN #1 #2 #3 #4 #5 {
1756   \hbox{
1757     \bool_if:NTF { #4 } {
1758       \vbox{
1759         \ifx\AMCnTextSign\@empty\@empty\else
1760           \hbox{\AMCnTextSign}\vspace{\AMCnumeric@Vspace}\fi
1761         \bool_if:NTF \amc_num_strict_bool {
1762           \int_compare:nNnTF { #5 } = 0 {
1763             \amc_num_sign_boxes:Nn { 1 } { #2 }
1764           }{
1765             \amc_num_sign_boxes:Nn { #5 } { #2 }
1766           }
1767         }{
1768           \amc_num_sign_boxes:Nn { #5 } { #2 }
1769         }
1770       }
1771       \hspace{.5em}
1772       \vrule
1773       \hspace{.5em}
1774     } { }
1775   \hbox{

```

```

1776     \bool_if:NTF \amc_num_vertical_bool
1777     \amc_num_integer_boxes_v:Nnn \amc_num_integer_boxes_h:Nnn
1778     #1 { #2 } { #3 }
1779   }
1780 }
1781 }
1782

```

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

```

1783
1784 \cs_new:Npn \amc_num_build_integer_scoring:Nnnnn #1 #2 #3 #4 #5 {
1785   \amc_num_check_score_opts:
1786   \tl_clear:N #1
1787   \int_set_eq:NN \amc_num_digit_int { #4 }
1788   \int_while_do:nNnn \amc_num_digit_int > 0 {
1789     \bool_if:NTF \amc_num_strict_bool {
1790       \AMCmessage{B=requires. #3
1791         \int_to_Alph:n \amc_num_digit_int = 1}
1792     } {
1793       \AMCmessage{B=default. #3
1794         \int_to_Alph:n \amc_num_digit_int =
1795         \bool_if:NTF \amc_num_ignoreblank_bool { } { 0 }
1796       }
1797     }
1798     \int_compare:nNnTF \amc_num_digit_int = #4 { } {
1799       \bool_if:NTF \amc_num_ignoreblank_bool {
1800         \tl_put_right:Nx #1 { ~.~ }
1801         \int_compare:nNnTF \amc_num_digit_int = #5 {
1802           \tl_put_right:Nx #1 { "." ~.~ }
1803         } { }
1804       } {
1805         \tl_put_left:Nn #1 { ( }
1806         \tl_put_right:Nx #1 { ) *
1807         \int_use:N \amc_num_base_int + }
1808       }
1809     }
1810     \tl_put_right:Nx #1
1811     { #3 \int_to_Alph:n \amc_num_digit_int }
1812     \int_decr:N \amc_num_digit_int
1813   }
1814   \bool_if:NTF \amc_num_ignoreblank_bool {
1815     \tl_put_left:Nn #1 { ( 0 + ( }
1816     \tl_put_right:Nn #1 { ) ) }
1817     \int_compare:nNnTF \amc_num_decd_int > 0 {
1818       \tl_put_right:Nx #1 { * ( 10 ** \int_eval:n { #5 } ) }
1819     } { }
1820   } {
1821     \tl_put_left:Nn #1 { ( }
1822     \tl_put_right:Nn #1 { ) }
1823   }
1824   \bool_if:NTF { #2 } {
1825     \bool_if:NTF \amc_num_strict_bool {

```

```

1826     \AMCmessage{B=requires. sign #3 =1}
1827   } {
1828     \AMCmessage{B=default. sign #3 =1}
1829   }
1830   \tl_put_right:Nx #1 { * ( sign #3 ) }
1831 } { }
1832 }
1833

```

Then the command `\AMCnumericShow{⟨x⟩}{⟨options⟩}` itself:

```

1834
1835 \fp_new:N \amc_num_result_fp
1836 \fp_new:N \amc_num_correct_fp
1837 \clist_new:N \amc_num_digits_clist
1838 \clist_new:N \amc_num_expo_digits_clist
1839 \int_new:N \amc_num_digit_int
1840 \int_new:N \amc_num_sign_int
1841 \int_new:N \amc_num_expo_sign_int
1842 \tl_new:N \amc_num_compute_tl
1843 \tl_new:N \amc_num_expo_tl
1844 \int_new:N \amc_num_correct_expo_int
1845
1846 \cs_new:Npn \amc_numeric_show:nn #1 #2 {

```

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

```

1847   \ifAMC@ensemble\ifAMCformulaire@dedans
1848     \AMCmessage{Q=\the\AMCid@quest}
1849   \fi\fi

```

Then we parse the options from `⟨opts⟩`:

```

1850   {\keys_set:nn { amcnumeric } { #2 }
1851     \bool_if:nTF { \bool_if_p:N\amc_num_significant_bool
1852       && \int_compare_p:n { \amc_num_base_int != 10 } } {
1853       \message{^^J!~AMCnumeric~Error:~significant=true~can't~be~used~with~base!=10.^^J}
1854     } {}
1855     \bool_if:nTF { \int_compare_p:n { \amc_num_expo_int != 0 }
1856       && \int_compare_p:n { \amc_num_base_int != 10 } } {
1857       \message{^^J!~AMCnumeric~Error:~scientific~notation~can't~be~used~with~base!=10.^^J}
1858     } {}

```

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

```

1859   \ifx\@empty#1\@empty
1860     \fp_set:Nn \amc_num_correct_fp { 0 }
1861     \fp_set:Nn \amc_num_mantissa_fp { 0 }
1862     \int_set:Nn \amc_num_correct_expo_int { 0 }
1863   \else
1864     \bool_if:nTF \amc_num_significant_bool {
1865       \amc_fp_n_significant_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_ndigits_int
1866     } {
1867       \int_compare:nNnTF \amc_num_expo_int > 0 {
1868         \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_correct_expo_int { #1 }
1869         \int_compare:nNnTF { \amc_num_ndigits_int - \amc_num_decd_int } > 1 {
1870           \fp_set:Nn \amc_num_mantissa_fp {
1871             \amc_num_mantissa_fp * 10^( \amc_num_ndigits_int - \amc_num_decd_int - 1 )

```

```

1872     }
1873     \int_set:Nn \amc_num_correct_expo_int {
1874         \amc_num_correct_expo_int - ( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1875     }
1876 } {}
1877 \amc_fp_n_digits:Nnn \amc_num_correct_fp \amc_num_mantissa_fp \amc_num_decd_int
1878 } {
1879     \amc_fp_n_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_decd_int
1880 }
1881 }
1882 \fi

```

Now extracts the required digits:

```

1883 \ifx\@empty#1\@empty
1884     \amc_invalid_digits:Nn \amc_num_digits_clist \amc_num_ndigits_int
1885     \amc_invalid_digits:Nn \amc_num_expo_digits_clist \amc_num_expo_int
1886     \int_set:Nn \amc_num_sign_int { 0 }
1887     \int_set:Nn \amc_num_expo_sign_int { 0 }
1888 \else
1889     \amc_fp_to_digits:Nnnn \amc_num_digits_clist \amc_num_correct_fp
1890     \amc_num_ndigits_int \amc_num_base_int
1891     \amc_get_fp_sign:Nn \amc_num_sign_int \amc_num_correct_fp
1892     \int_compare:nNnTF \amc_num_expo_int > 0 {
1893         \amc_fp_to_digits:Nnnn \amc_num_expo_digits_clist \amc_num_correct_expo_int
1894         \amc_num_expo_int \amc_num_base_int
1895         \amc_get_int_sign:Nn \amc_num_expo_sign_int \amc_num_correct_expo_int
1896     } {}
1897 \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.

```

1898 \fp_set:Nn \amc_num_result_fp { #1 }
1899 \AMCmessage{B=numval=\fp_to_scientific:N \amc_num_result_fp ,
1900     numex=\int_use:N \amc_num_exact_int,
1901     numapp=\int_use:N \amc_num_approx_int,
1902     numsex=\AMC@numeric@scoreexact,
1903     numsapp=\AMC@numeric@scoreapprox
1904 }
1905 \bool_if:NNTF \amc_num_scoring_bool {
1906     \AMCmessage{B=haut=mz=d=undef,p=undef,
1907         formula=(Vdifference <= \int_use:N \amc_num_exact_int ?
1908             \AMC@numeric@scoreexact :
1909             \int_compare:nNnTF \amc_num_approx_int = 0 {
1910                 \AMC@numeric@scorewrong
1911             } {
1912                 (Vdifference <= \int_use:N \amc_num_approx_int ?
1913                     \AMC@numeric@scoreapprox : \AMC@numeric@scorewrong)
1914             }
1915         )}
1916 } {}
1917 \amc_num_build_integer_scoring:Nnnnn
1918     \amc_num_compute_tl \amc_num_sign_bool { digit } \amc_num_ndigits_int
1919     \amc_num_decd_int
1920 \int_compare:nNnTF \amc_num_expo_int > 0 {

```



```

1921 \amc_num_build_integer_scoring:Nnnnn
1922 \amc_num_expo_tl \amc_num_exposign_bool { expo } \amc_num_expo_int { 0 }
1923 \AMCmessage{B= set. intE = \amc_num_expo_tl}
1924 } {}
1925 \AMCmessage{B= set.intV = \fp_to_int:N\amc_num_correct_fp ,
1926 set.intXX = \amc_num_compute_tl }
1927 \int_compare:nNnTF \amc_num_expo_int > 0 {
1928 \AMCmessage{B= set.intX = intXX * \int_use:N\amc_num_base_int ** ( intE - (\int_use:N\amc_num_correc
1929 }{
1930 \AMCmessage{B= set.intX = intXX}
1931 }
1932 \int_compare:nNnTF \amc_num_expo_int > 0 {
1933 \AMCmessage{B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (intE - \int_use:N\amc_num_decd
1934 }{
1935 \AMCmessage{B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (- \int_use:N\amc_num_decd_int))}
1936 }
1937 \ifx\@empty\AMC@numeric@keepas\@empty\else
1938 \AMCmessage{B= setglobal.\AMC@numeric@keepas = valueX}
1939 \fi
1940 \ifx\@empty#1\@empty
1941 \bool_if:NNTF \amc_num_significant_bool {
1942 \AMCmessage{B=set.Vdifference=0}
1943 }{
1944 \ifx\@empty\AMC@numeric@alsocorrect\@empty
1945 \AMCmessage{B=set.Vdifference=0}
1946 \else
1947 \AMCmessage{B="set.Vdifference =
1948 amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\amc_r
1949 "}
1950 \fi
1951 }
1952 \else
1953 \bool_if:NNTF \amc_num_significant_bool {
1954 \AMCmessage{B=set.Vdifference="min( abs((intV)-(intX)) ,
1955 abs(\int_use:N\amc_num_base_int * (intV) - (intX)) ,
1956 abs((intV) - \int_use:N\amc_num_base_int * (intX)) )"}
1957 } {
1958 \ifx\@empty\AMC@numeric@alsocorrect\@empty
1959 \AMCmessage{B=set.Vdifference=abs((intV)-(intX))}
1960 \else
1961 \AMCmessage{B="set.Vdifference =
1962 min( amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\amc_r
1963 abs((intV)-(intX)) )"}
1964 \fi
1965 }
1966 \fi

```

Begin now with the frame around all the boxes:

```

1967 \ifAMC@extractOnly\else
1968 \vspace{1.5ex}\par{
1969 \fboxrule=\AMCncol@BorderWidth
1970 \fcolorbox{\AMCncol@Border}{\AMCncol@Background}{
1971 \bool_if:NNTF \amc_num_expovetical_bool {
1972 \hbox{\vbox{

```

```

1973         \vbox{\amc_num_integer_boxes:NnnNN
1974         \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1975         \amc_num_sign_int}
1976         \int_compare:nNnTF \amc_num_expo_int > 0 {
1977             \vspace{\AMCnumeric@Vspace}
1978             \vbox{\hbox{\AMCexponent}}
1979             \vspace{\AMCnumeric@Vspace}
1980             \vbox{\amc_num_integer_boxes:NnnNN
1981                 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
1982                 \amc_num_expo_sign_int}
1983         } {}
1984     }}
1985 } {
1986     \amc_num_integer_boxes:NnnNN
1987     \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
1988     \amc_num_sign_int
1989     \int_compare:nNnTF \amc_num_expo_int > 0 {
1990         \hspace{\AMCnumeric@Hspace}\AMCexponent\hspace{\AMCnumeric@Hspace}
1991         \amc_num_integer_boxes:NnnNN
1992         \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
1993         \amc_num_expo_sign_int
1994     } {}
1995 }
1996 }
1997 }
1998 \fi

```

And tell AMC that we finished with this question:

```

1999 \ifAMC@ensemble\else\vspace{\AMCpostNquest}\par\fi
2000 \ifAMC@ensemble\ifAMCformulaire@dedans
2001     \AMCmessage{FQ}
2002 \fi\fi
2003 }
2004 }
2005
2006 \cs_new_eq:NN \AMCnumericShow \amc_numeric_show:nn
2007

```

`\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.

```

2008 \cs_new:Npn \amc_numeric_hide:nn #1 #2 {
2009     \keys_set:nn { amcnumeric } { #2 }
2010     \AMCncontextGoto
2011     \ifAMC@qbloc\else\vspace{1.5ex}\par\fi
2012 }
2013
2014 \cs_new_eq:NN \AMCnumericHide \amc_numeric_hide:nn
2015
2016 \ExplSyntaxOff
2017 \def\AMCnumericChoicesPlain{%
2018     \AMC@if@separate@question{\AMC@mem@category{numeric}}}%
2019     \AMCformatChoices{\AMCnumericShow}{\AMCnumericHide}%
2020 }

```

The $\langle value \rangle$ argument is often given as a macro, that is to be expanded before calling `\AMCNumericChoicesPlain`, so that its value will be the same in the separate answer sheet...

```
2021 \ExplSyntaxOn
2022
2023 \cs_new:Npn \amc_numeric_choices:nn #1#2 {
2024   \AMCNumericChoicesPlain{#1}{#2}
2025 }
2026 \cs_generate_variant:Nn \amc_numeric_choices:nn { xn }
2027 \cs_new_eq:NN \AMCNumericChoices \amc_numeric_choices:xn
2028
2029 \ExplSyntaxOff
```

4.13.3 Intervals

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

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

As an example,

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

produces (in correction mode):

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

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

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

```
2030 \def\AMCIntervalFormat#1#2{[#1,\,#2]}
```

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

```
2031 \ExplSyntaxOn
2032
2033 \fp_new:N \amc_interv_a
2034 \fp_new:N \amc_interv_b
2035 \cs_new:Npn \amc_intervals:nnnn #1 #2 #3 #4 {
2036   \fp_set:Nn \amc_interv_a { #2 }
2037   \fp_do_while:nn { \amc_interv_a < #3 } {
2038     \fp_set:Nn \amc_interv_b { \amc_interv_a + #4 }
2039     \fp_compare:nTF { \amc_interv_a <= #1 < \amc_interv_b }
2040       \correctchoice \wrongchoice
2041     {\AMCIntervalFormat{\fp_use:N \amc_interv_a}{\fp_use:N \amc_interv_b}}
2042     \fp_set:Nn \amc_interv_a \amc_interv_b
```

```

2043   }
2044 }
2045 \cs_new_eq:NN \AMCIntervals \amc_intervals:nmmn
2046
2047 \ExplSyntaxOff

```

4.14 Open questions

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

As an example,

```

\begin{question}{Linux}
  What is the first name of the person who started working on the Linux kernel?
  \AMCOpen{}{\wrongchoice[w]{w}\scoring{0}\correctchoice[c]{c}\scoring{2}}
\end{question}

```

shows:

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

☐ w ☐ c

.....

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

Begin with the options definitions:

```

2048 \def\AMCotextGoto{}
2049 \def\AMCotextReserved{}
2050 \def\AMCocol@Background{lightgray}
2051 \def\AMCocol@BoxFrameRule{white}
2052 \def\AMCocol@FrameRule{black}
2053 \def\AMCocol@Foreground{}
2054 \def\AMCopen@answer{}
2055 \def\AMCopen@question{}
2056 \def\AMCopen@lineuptext{}
2057 \define@key{AMCOpen}{backgroundcol}{\def\AMCocol@Background{#1}}
2058 \define@key{AMCOpen}{foregroundcol}{\def\AMCocol@Foreground{#1}}
2059 \define@key{AMCOpen}{Treserved}{\def\AMCotextReserved{#1}}
2060 \define@key{AMCOpen}{question}[\AMCid@name]{\def\AMCopen@question{#1}}
2061 \define@key{AMCOpen}{answer}{\def\AMCopen@answer{#1}}
2062 \define@key{AMCOpen}{contentcommand}[AMCopen@lines]{\def\AMCopen@contentcommand{#1}}
2063 \newdimen\AMCopen@Hspace\AMCopen@Hspace=.5em
2064 \define@key{AMCOpen}{hspace}{\AMCopen@Hspace=#1}
2065 \def\AMCopen@Width{.95\linewidth}
2066 \define@key{AMCOpen}{width}{\def\AMCopen@Width{#1}}
2067 \newdimen\AMCopen@LineHeight\AMCopen@LineHeight=1cm
2068 \define@key{AMCOpen}{lineheight}{\AMCopen@LineHeight=#1}

```

```

2069 \newcount\AMCopen@Lines\AMCopen@Lines=1
2070 \define@key{AMCOpen}{lines}{\AMCopen@Lines=#1}
2071 \newdimen\AMCopen@boxmargin\AMCopen@boxmargin=3pt
2072 \define@key{AMCOpen}{boxmargin}{\AMCopen@boxmargin=#1}
2073 \newdimen\AMCopen@boxframerule\AMCopen@boxframerule=1pt
2074 \define@key{AMCOpen}{boxframerule}{\AMCopen@boxframerule=#1}
2075 \define@key{AMCOpen}{boxframerulecol}{\def\AMCocol@BoxFrameRule{#1}}
2076 \define@key{AMCOpen}{framerulecol}{\def\AMCocol@FrameRule{#1}}
2077 \newdimen\AMCopen@framerule\AMCopen@framerule=1pt
2078 \define@key{AMCOpen}{framerule}{\AMCopen@framerule=#1}
2079 \define@key{AMCOpen}{lineuptext}{\def\AMCopen@lineuptext{#1}}
2080 \define@boolkey{AMCOpen}{dots}[true]{}
2081 \define@boolkey{AMCOpen}{scan}[true]{}
2082 \define@boolkey{AMCOpen}{retick}[true]{}
2083 \define@boolkey{AMCOpen}{annotate}[false]{}
2084 \define@boolkey{AMCOpen}{lineup}[false]{}
2085 \setkeys{AMCOpen}{dots,scan,retick,annotate,lineup,contentcommand}
2086 \newcommand\AMCopenOpts[1]{\setkeys{AMCOpen}{#1}}

```

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

```

2087 \newcommand\AMCopen@lines{%
2088   \begin{minipage}{\AMCopen@Width}%
2089     \loop\vspace{\AMCopen@LineHeight}
2090     \hspace*{.5em}\ifAMC@correc\smash{\AMCopen@answer}\def\AMCopen@answer{}\fi%
2091     \ifKV@AMCOpen@dots%
2092     \dotfill\hspace*{.5em}%
2093     \fi
2094     \ifnum\AMCopen@Lines>\@ne\par\advance\AMCopen@Lines\m@ne\repeat%
2095   \end{minipage}%
2096 }
2097 \newcommand\AMCopenShow[2]{
2098   \ifAMC@ensemble\ifAMC@formulaire@dedans%
2099     \AMCmessage{Q=\the\AMCid@quest}%
2100   \fi\fi%
2101   {\setkeys{AMCOpen}{#1}%
2102     \ifKV@AMCOpen@lineup%
2103       \ifAMC@ensemble\else%
2104         \ifx\@empty\AMCopen@lineuptext\@empty\fi%
2105       \fi%
2106       \ifAMC@correc\smash{\AMCopen@answer}\fi%
2107       \ifx\@empty\AMCopen@lineuptext\@empty%
2108         \dotfill%
2109       \else%
2110         \AMCopen@lineuptext\hfill%
2111       \fi%
2112     \else%
2113       \hspace*{.5em}\linebreak[1]\hspace*{\fill}%
2114     \fi%
2115   }\AMCnoCompleteMulti%
2116   \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2117   \def\AMCanswer##1##2{\ifAMC@ensemble ##1\else%
2118     \ifAMC@inside@box ##1\else\AMCboxOutsideLetter{##1}{##2}\fi\fi%
2119     \hspace{\AMCopen@Hspace}}%

```

```

2120 \fboxsep=\AMCopen@boxmargin%
2121 \fboxrule=\AMCopen@boxframerule%
2122 \fcolorbox{\AMCocol@BoxFrameRule}{\AMCocol@Background}{%
2123 \ifAMC@ensemble\AMCopen@question%
2124 \ifx\@empty\AMCopen@question\@empty\else\hspace{\AMCopen@Hspace}\fi%
2125 \fi%
2126 \begin{choicescustom}[o]%
2127 \ifx\AMCocol@Foreground\@empty\@empty\else%
2128 \def\AMC@boxcolor{\AMCocol@Foreground}%
2129 \fi%
2130 #2%
2131 \ifKV@AMCopen@scan\else\AMCdontScan\fi%
2132 \ifKV@AMCopen@retick\AMCreTick\fi%
2133 \ifKV@AMCopen@annotate\else\AMCdontAnnotate\fi%
2134 \end{choicescustom}%
2135 \ifx\@empty\AMCotextReserved\@empty%
2136 \hspace{-\AMCopen@Hspace}%
2137 \else%
2138 \ifx\AMCocol@Foreground\@empty\@empty%
2139 \AMCotextReserved%
2140 \else%
2141 \textcolor{\AMCocol@Foreground}{\AMCotextReserved}%
2142 \fi%
2143 \fi%
2144 }}%
2145 \ifKV@AMCopen@lineup\else%
2146 \par\nobreak\noindent%
2147 \hspace*{\fill}{%
2148 \fboxrule=\AMCopen@framerule%
2149 \fcolorbox{\AMCocol@FrameRule}{white}{%
2150 \csname\AMCopen@contentcommand\endcsname
2151 }}%
2152 \vspace{\AMCpostOquest}\par%
2153 \fi%
2154 }%
2155 \ifAMC@ensemble\ifAMCformulaire@dedans%
2156 \AMCmessage{FQ}%
2157 \fi\fi%
2158 }
2159 \newcommand\AMCopenHide[2]{%
2160 \AMCotextGoto%
2161 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2162 }
2163 \def\AMCopen{%
2164 \AMC@if@separate@question{\AMC@mem@category{open}}%
2165 \AMCformatChoices{\AMCopenShow}{\AMCopenHide}%
2166 }

```

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}

2167 \def\AMCbotextGoto{}
2168 \def\AMCbo@help{}
2169 \define@key{AMCBoxOnly}{help}{\def\AMCbo@help{#1}}
2170 \define@boolkey{AMCBoxOnly}{ordered}[false]{}
2171 \setkeys{AMCBoxOnly}{ordered}
2172 \newcommand\AMCboOpts[1]{\setkeys{AMCBoxOnly}{#1}}
2173 \newcommand\AMCboShow[2]{%
2174   \ifAMC@ensemble\ifAMCformulaire@dedans%
2175     \AMCmessage{Q=\the\AMCid@quest}%
2176   \fi\fi%
2177   {\setkeys{AMCBoxOnly}{#1}%
2178     \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2179     \def\AMCanswer##1##2{\hspace{\AMCformHSpace} \ifAMC@ensemble ##1\else%
2180       \ifAMC@inside@box ##1\else{\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2181     }%
2182     \ifAMC@ensemble\AMCbo@help\fi%
2183     \ifKV@AMCBoxOnly@ordered%
2184       \begin{choicescustom}[o]%
2185     \else%
2186       \begin{choicescustom}%
2187     \fi%
2188     #2
2189     \end{choicescustom}%
2190   }%
2191   \ifAMC@ensemble\ifAMCformulaire@dedans%
2192     \AMCmessage{FQ}%
2193   \fi\fi%
2194 }
2195 \newcommand\AMCboHide[2]{
2196   \AMCbotextGoto%
2197   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2198 }
2199 \def\AMCBoxOnly{%
2200   \AMC@if@separate@question{\AMC@mem@category{box}}%
2201   \AMCformatChoices{\AMCboShow}{\AMCboHide}%
2202 }

```

4.16 Page formatting

4.16.1 Watermark

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

```

2203 \DeclareFontShape{OT1}{cmr}{b}{n}{<35->cmr17}{-}
2204 \def\AMC@watertext{\AMC@loc@draft}
2205 \newcommand\AMCw@termark{%

```

```

2206 \setlength{\@tempdimb}{.5\paperwidth}%
2207 \setlength{\@tempdimc}{-.5\paperheight}%
2208 \put(\strip@pt\@tempdimb,\strip@pt\@tempdimc){%
2209   \makebox(0,0){\rotatebox{45}{\AMC@LR{%
2210     \textcolor[gray]{0.8}{
2211       \fontencoding{OT1}\fontfamily{cmr}
2212       \fontseries{b}\fontshape{n}
2213       \fontsize{90pt}{120pt}
2214       \selectfont
2215       \AMC@watertext}}}}}}
2216 \newcommand\AMCwaterprint[1]{%
2217   \setbox\@tempboxa\vbox to \z@{%
2218     \vbox{%
2219       \hbox to \z@{%
2220         #1\hss}}\vss}
2221   \dp\@tempboxa\z@
2222   \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}.

```

2223 \def\AMCcercle#1#2{%
2224   {\setlength{\unitlength}{1mm}%
2225     \begin{picture}(\#1,\#1)(-#2,-#2)\thinlines\circle*{\#1}\end{picture}}}
2226 \def\m@rqueCalage{\AMCcercle{3.6}{1.8}}
2227 \def\m@rque#1{\AMC@tracebox{1}{\#1}{\m@rqueCalage}}
2228 \def\he@dtaille#1{% \par cancels the \leavevmode
2229   % introduced by https://github.com/pietvo/fancyhdr/commit/6b1ad10eeb5bc3d804f3cd2cf193e6440d0229e6
2230   \par\vbox to 1cm{\#1}}
2231 \def\he@dbas#1{\he@dtaille{\vspace*{\fill}\#1}}
2232 \def\he@dhaut#1{\he@dtaille{\#1\vspace*{\fill}}}}
2233 \def\AMC@intituleHead{\AMC@loc@corrected}
2234 \def\AMC@note{}
2235 \def\AMCsetFoot#1{\def\AMC@note{\#1}}
2236 \newcommand\AMCStudentNumber{\the\AMCid@etud}
2237 \newcommand\AMCIDBoxesA{\AMC@binaryCode{id=1,ndigits=\AMC@NCBetud}{\the\AMCid@etud}}
2238 \newcommand\AMCIDBoxesB{\AMC@binaryCode{id=2,ndigits=\AMC@NCBpage}{\thepage}}
2239 \newcommand\AMCIDBoxesC{\AMC@binaryCode{id=3,ndigits=\AMC@NCBcheck}{\the\AMCid@check}}
2240 \newcommand\AMCIDBoxesABC{%
2241   \hbox{\vbox{\noindent\AMCIDBoxesA\
2242     \noindent\AMCIDBoxesB\AMCIDBoxesC}}}%
2243 }
2244 \AtBeginPage{\ifAMC@pagelayout\global\advance\AMCid@check\m@ne%
2245   \ifnum\AMCid@check<1\global\AMCid@check=\AMCid@checkmax\fi%
2246   \AMC@pagepos%
2247   \ifAMC@watermark\ifAMC@correthead\else\AMCwaterprint{\AMCwatermark}%

```



```

2248 \fi\fi\fi}
2249 \fancypagestyle{AMCpageHeadOnly}{%
2250 \fancyhf{}\fancyhead[C]{\textsc{\AMC@intituleHead}}%
2251 \renewcommand{\headrulewidth}{0pt}%
2252 \renewcommand{\footrulewidth}{0pt}%
2253 }
2254 \fancypagestyle{AMCpageFull}{%
2255 \fancyhf{}%
2256 \fancyhead[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHG}}}}%
2257 \fancyhead[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHD}}}}%
2258 \fancyfoot[L]{\AMC@LR{\leavevmode\m@rque{positionBG}}}%
2259 \fancyfoot[R]{\AMC@LR{\leavevmode\m@rque{positionBD}}}%
2260 \fancyhead[C]{\AMC@LR{\he@dhaut{%
2261 \begin{minipage}[b]{\AMC@CBtaille}\AMCboxColor{black}%
2262 \ifAMCids@top\vbox to \AMCids@height{\texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2263 \AMCIDBoxesABC
2264 \end{minipage}%
2265 \ifAMCids@side\hbox to \AMCids@width{\hspace*{\fill}%
2266 \texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2267 }}}%
2268 \fancyhfoffset[EOLR]{5mm}%
2269 \fancyfoot[C]{\AMC@note}%
2270 \renewcommand{\headrulewidth}{0pt}%
2271 \renewcommand{\footrulewidth}{0pt}%
2272 }
2273 \newcommand\AMCsubjectPageTag{%
2274 \fbox{\texttt{\the\AMCid@etud:\thepage}}%
2275 }
2276 \fancypagestyle{AMCpageNoMarks}{%
2277 \fancyhf{}%
2278 \fancyhead[R]{\AMCsubjectPageTag}%
2279 \fancyfoot[C]{\AMC@note}%
2280 \renewcommand{\headrulewidth}{0pt}%
2281 \renewcommand{\footrulewidth}{0pt}%
2282 }
2283 \fancypagestyle{AMCpageEmpty}{%
2284 \fancyhf{}%
2285 \renewcommand{\headrulewidth}{0pt}%
2286 \renewcommand{\footrulewidth}{0pt}%
2287 }
2288 \AtBeginDocument{%
2289 \ifAMC@pagelayout%
2290 \ifAMC@correthead
2291 \pagestyle{AMCpageHeadOnly}
2292 \else
2293 \pagestyle{AMCpageFull}
2294 \fi
2295 \fi
2296 }

```

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

```

2297 \newcommand{\onecopy}[2]{%
2298   \ifx\AMCNombreCopies\undefined\AMCnum@copies=#1%
2299   \else\AMCnum@copies=\AMCNombreCopies\fi%
2300   \AMCmessage{TOTAL=\the\AMCnum@copies}%
2301   \message{^^JAMC:copies:total=\the\AMCnum@copies^^J}%
2302   \AMCid@etud=\AMCid@etudstart%
2303   \ifnum\AMCid@etud=0\AMCid@etud=\AMC@premierecopie\fi%
2304   \AMCid@etudfin=\AMCnum@copies%
2305   \advance\AMCid@etudfin\AMCid@etud\relax%
2306   \ifAMC@correthead\AMCid@etudfin=\AMC@premierecopie%
2307   \message{^^JAMC:copies:total=1^^J}%
2308   \fi
2309   \ifAMC@pdfform\begin{Form}\fi%
2310   \loop{%
2311     \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
2312       \string\rngstate{\the\AMCid@etud}\the\AMC@SR}%
2313     }\fi%
2314     \AMC@zoneformulairefalse\setcounter{page}{1}\setcounter{section}{0}%
2315     \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageNoMarks}\fi\fi%
2316     \AMCnumero{1}%
2317     \ifAMC@calibration\AMCmessage{ETU=\the\AMCid@etud}\fi%
2318     \AMC@multiclear%
2319     \global\AMC@keepmemoryfalse%
2320     #2%
2321     \ifAMC@keepmemory\else\AMC@mem@clear\fi%
2322     \clearpage}%
2323     \message{^^JAMC:copies:add=1^^J}%
2324     \advance\AMCid@etud\@ne\ifnum\AMCid@etud<\AMCid@etudfin\repeat%
2325     \global\AMCid@etudstart=\AMCid@etud%
2326     \ifAMC@pdfform\end{Form}\fi%
2327     \AMC@multi@report%
2328 }

```

`\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.

```

2329 \newcount\AMC@addpages
2330 \newcommand{\AMCaddpagesto}[1]{%
2331   \AMC@addpages=#1\advance\AMC@addpages\@ne%
2332   \clearpage%
2333   \@whilenum\thepage<\AMC@addpages\do{%
2334     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2335     \hbox{}\clearpage%
2336   }%
2337 }

```

`\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).

```
2338 \def\AMCcleardoublepage{%
2339   \clearpage%
2340   \ifodd\thepage\else%
2341     \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2342     \hbox{}\clearpage%
2343   \fi%
2344 }
```

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

```
2345 \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 2346 \newcommand\AMCstudentlabel[1]{\the\AMCid@etud-#1}
2347 \def\AMClabel#1{\expandafter\label{\AMCstudentlabel{#1}}}
2348 \def\AMCref#1{\expandafter\ref{\AMCstudentlabel{#1}}}
2349 \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.

```
2350 \newcommand{\AMCqlabel}[1]{%
2351   \AMClabel{#1}%
2352 }
```

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.

```
2353 \newcommand{\AMCassociation}[2][]{%
2354   \ifAMC@calibration%
2355     \immediate\write\AMC@XYFILE{\string\association{\the\AMCid@etud}{#2}{#1}}%
2356   \fi%
2357 }
```

`\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>}`.

```
2358 \newcommand{\AMCstudentslistfile}[2]{%
2359   \ifAMC@calibration%
2360     \immediate\write\AMC@XYFILE{\string\with{studentslistfile=#1}}%
2361     \immediate\write\AMC@XYFILE{\string\with{studentslistkey=#2}}%
2362   \fi%
2363 }
```

4.19 Package options

See section 3.1 for the options descriptions.

```
2364 \def\AMC@lang@code{}
```

```

2365 \DeclareOptionX{noshuffle}{\AMC@ordretrue}
2366 \DeclareOptionX{noshufflegroups}{\AMC@shuffleGfalse}
2367 \DeclareOptionX{fullgroups}{\AMC@fullGroupstrue}
2368 \DeclareOptionX{answers}{\AMC@corretheadtrue\AMC@correcttrue}
2369 \DeclareOptionX{indivanswers}{\AMC@correcttrue}
2370 \DeclareOptionX{textpos}{\AMC@textPostrue}
2371 \DeclareOptionX{extractonly}{\AMC@extractOnlytrue\AMC@textPostrue\AMC@boxStyle{shape=none}\AMC@boxedAnswer}
2372 \DeclareOptionX{box}{\AMC@qbloctrue}
2373 \DeclareOptionX{asbox}{\AMC@asqbloctrue}
2374 \DeclareOptionX{separateanswersheet}{\AMC@ensembletrue}
2375 \DeclareOptionX{digits}{\AMC@inside@digittrue}
2376 \DeclareOptionX{ordre}{\AMC@ordretrue}
2377 \DeclareOptionX{correc}{\AMC@corretheadtrue\AMC@correcttrue}
2378 \DeclareOptionX{modele}{\AMC@corretheadtrue\AMC@correcfalse\AMC@ordretrue}
2379 \DeclareOptionX{correcindiv}{\AMC@correcttrue}
2380 \DeclareOptionX{init}{\AMC@SR@time}
2381 \DeclareOptionX{bloc}{\AMC@qbloctrue}
2382 \DeclareOptionX{completemulti}{\AMC@complete@multitrue}
2383 \DeclareOptionX{insidebox}{\AMC@inside@boxtrue}
2384 \DeclareOptionX{ensemble}{\AMC@ensembletrue}
2385 \DeclareOptionX{chiffres}{\AMC@inside@digittrue}
2386 \DeclareOptionX{outsidebox}{\AMC@outside@boxtrue}
2387 \DeclareOptionX{calibration}{\AMC@calibrationtrue}
2388 \DeclareOptionX{nowatermark}{\AMC@watermarkfalse}
2389 \newcommand\AMC@catalogMode{%
2390   \AMC@catalogtrue%
2391   \AMC@watermarkfalse\AMC@corretheadtrue%
2392   \AMC@correcttrue\AMC@ordretrue\AMC@shuffleGfalse%
2393   \AMC@fullGroupstrue%
2394   \def\AMC@intituleHead{\AMC@loc@catalog}\AMC@affichekeystrue}
2395 \newcommand\AMC@keys@next{\AMC@keys@linefalse}
2396 \newcommand\AMC@keys@line{\AMC@keys@linetrue}
2397 \DeclareOptionX{catalog}{\AMC@catalogMode}
2398 \DeclareOptionX{keys}[next]{\csname AMC@keys@#1\endcsname{}}
2399 \DeclareOptionX{francais}{\def\AMC@lang@code{FR}\AMC@loc@FR}
2400 \DeclareOptionX{lang}{\def\AMC@lang@code{#1}\csname AMC@loc@#1\endcsname}
2401 \DeclareOptionX{versionA}{%
2402   \def\AMC@id@checkmax{31}\def\AMC@NCBetud{9}\def\AMC@NCBpage{4}%
2403   \def\AMC@NCBcheck{5}\setlength{\AMC@CBtaille}{4cm}%
2404   \def\AMC@premierecopie{100}}
2405 \DeclareOptionX{plain}{\AMC@plaintrue}
2406 \DeclareOptionX{nopage}{\AMC@pagelayoutfalse}
2407 \DeclareOptionX{postcorrect}{\AMC@postcorrecttrue}
2408 \DeclareOptionX{automarks}{\AMC@automarkstrue}
2409 \newif\ifAMCneeds@storebox\AMCneeds@storeboxfalse
2410 \DeclareOptionX{storebox}{\AMCneeds@storeboxtrue}
2411 \DeclareOptionX{pdfform}{\AMC@pdfformtrue}
2412 \DeclareOptionX{codedigit}{\AMC@codeID@{#1}}
2413 \newif\ifAMC@survey\AMC@surveyfalse
2414 \DeclareOptionX{survey}{\AMC@surveytrue}
2415 \ProcessOptionsX
2416 \ifAMCneeds@storebox
2417   \RequirePackage{storebox}\AtBeginDocument{{}}%

```

```

2418 \fi
2419 \ifAMC@pdfform
2420   \AMCmessage{VAR:project:pdfform=1}%
2421   \AMCboxStyle{shape=form}%
2422   \RequirePackage[pageanchor=false]{hyperref}%
2423 \else%
2424   \AMCmessage{VAR:project:pdfform=0}%
2425 \fi
2426 \AtBeginDocument{%
2427   \ifAMCneeds@storebox%
2428     \let\AMC@new@savebox=\newstorebox%
2429     \let\AMC@save@box=\storebox%
2430     \let\AMC@use@box=\usestorebox%
2431   \fi%
2432   \AMC@new@savebox{\AMC@ovalbox@R}%
2433   \AMC@new@savebox{\AMC@ovalbox@RF}%
2434   \AMC@new@savebox{\AMC@ovalbox@}%
2435   \AMC@new@savebox{\AMC@ovalbox@F}%
2436   \AMC@shapeprepare%
2437 }

```

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 (env.)
  auto (env.) 2438 \ifAMC@survey
question-auto (env.) 2439 \NewEnviron{Questionnaires}[1]{
  values (env.) 2440 \onecopy{#1}{
  values-auto (env.) 2441 \BODY
variable-auto (env.) 2442 }
  \answer 2443 }
  2444 \NewEnviron{auto}[1]{
  2445 \csvreader[head to column names, separator=tab]{#1}{}{
  2446 \BODY
  2447 }
  2448 }
  2449 \NewEnviron{question-auto}[3]{
  2450 \csvreader[head to column names, separator=tab]{#1}{}{
  2451 \ifcsvstrcmp{#2}{#3}{\BODY \\\}
  2452 }
  2453 }
  2454 \newenvironment{values}{}{}
  2455 \NewEnviron{values-auto}[5]{
  2456 \csvreader[head to column names, separator=tab]{#1}{}{
  2457 \ifcsvstrcmp{#2}{#3}{
  2458 \ifcsvstrcmp{#4}{#5}{\BODY \\\}
  2459 }{}
  2460 }{}
  2461 }
  2462 }

```

```

2463 \NewEnviron{variable-auto}[3]{
2464   \foreach \x in {#3}{
2465     \csvreader[head to column names, separator=tab]{#1}{}{
2466       \ifcsvstrcmp{#2}{\x}{\BODY}{}
2467     }
2468   }
2469 }
2470 \newcommand{\answer}[5][\global\advance\AMCrep@count\@ne\relax%
2471   \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%
2472   \global\AMCune@bonnettrue%
2473   \AMCload@@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkbox}%
2474     \else\AMC@box{#1}}\fi}{#2}{#3}{#4}{#5}}{\the\AMCrep@count}\ignorespaces}
2475 \RequirePackage{tikz}
2476 \usetikzlibrary{positioning, shapes, arrows, tikzmark, decorations.pathreplacing}
2477 \tikzset{
2478   checkbox-sc/.style={
2479     right=of lab\thecsvrow
2480   },
2481   vallab-sc/.style={
2482     text width=4cm,
2483     align=left,
2484   },
2485   checkbox-mc/.style={
2486   },
2487   vallab-mc/.style={
2488     above=of box\thecsvrow,
2489     text width=1.4cm,
2490     align=center,
2491   },
2492   varlab-mc/.style={
2493     text width=4cm,
2494     align=left,
2495   },
2496   node distance= 0mm,
2497 }
2498 \fi

```

4.21 Package Errors

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

```

2499 \def\AMC@error@explain{\PackageError{automultiplechoice}{
2500   Command \protect\explain\space can only be used inside\MessageBreak question like environments}{Some
2501 }}

```

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.

```

2502 \ifAMC@plain
2503 \else
2504   \IfFileExists{environ.sty}{\RequirePackage{environ}}{}
2505   \ifx\TeXversion\@undefined

```

```

2506 \else
2507   \RequirePackage{etex}
2508 \fi
2509 \fi

```

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

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

```

2510 \@ifpackageloaded{environ}{%
2511   \ifx\NewEnviron\undefined\PackageWarning{automultiplechoice}%
2512     {Package environ loaded but too old version:
2513      environnement examcopy/copieexamen will NOT be defined.}%
2514   \else\NewEnviron{examcopy}[1][5]{\onecopy{#1}{\BODY}}\fi}%
2515 {\PackageWarning{automultiplechoice}%
2516   {Package environ not loaded: environnement
2517    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.

```

2518 \ExplSyntaxOn
2519
2520 \cs_if_exist:NTF \pdfsavepos { } {
2521   \cs_if_exist:NTF \savepos { \cs_new_eq:NN \pdfsavepos \savepos } { }
2522 }
2523 \cs_if_exist:NTF \pdflastxpos { } {
2524   \cs_if_exist:NTF \lastxpos { \cs_new_eq:NN \pdflastxpos \lastxpos } { }
2525 }
2526 \cs_if_exist:NTF \pdflastypos { } {
2527   \cs_if_exist:NTF \lastypos { \cs_new_eq:NN \pdflastypos \lastypos } { }
2528 }

```

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

```

2529
2530 \cs_if_exist:NTF \pdfpagewidth { } {
2531   \cs_new_eq:NN \pdfpagewidth \paperwidth
2532 }
2533 \cs_if_exist:NTF \pdfpageheight { } {
2534   \cs_new_eq:NN \pdfpageheight \paperheight
2535 }
2536
2537 \ExplSyntaxOff

```

4.24 External control

`\SujetExterne` Some of the package options can be controlled defining `\xxxExterne` commands. For example,
`\ScoringExterne` the following command will format the subject document, whatever options are used in the \LaTeX
`\CorrigeExterne`
`\CorrigeIndivExterne`
`NoWatermarkExterne`

file:

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

2538 \ifx\SujetExterne\undefined\else
2539 \message{***SUJET***^^J}
2540 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse
2541 \fi
2542 \ifx\ScoringExterne\undefined\else
2543 \message{***SCORING***^^J}
2544 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse\AMC@invisibletrue
2545 \fi
2546 \ifx\CorrigeExterne\undefined\else
2547 \message{***CORRIGE***^^J}
2548 \AMC@calibrationfalse\AMC@corretheadtrue\AMC@correcttrue\AMC@watermarkfalse
2549 \fi
2550 \ifx\CorrigeIndivExterne\undefined\else
2551 \message{***CORRIGE***^^J}
2552 \AMC@calibrationfalse\AMC@corretheadfalse\AMC@correcttrue\AMC@watermarkfalse
2553 \fi
2554 \ifx\CatalogExterne\undefined\else
2555 \message{***CATALOG***^^J}
2556 \AMC@catalogMode
2557 \fi
2558 \ifx\NoWatermarkExterne\undefined\else
2559 \AMC@watermarkfalse
2560 \fi
2561 \ifx\codeDigitExterne\undefined\else
2562 \AMC@codeID@{\codeDigitExterne}
2563 \fi
```

4.25 Page layout

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

```
2564 \@ifpackageloaded{geometry}{\usepackage{geometry}}
2565 \ifAMC@pagelayout
2566   \ifAMC@correthead
2567     \geometry{hmargin=3cm,vmargin={1cm,1cm},includeheadfoot,headheight=1cm,footskip=1cm}
2568   \else
2569     \geometry{hmargin=3cm,headheight=2cm,headsep=.3cm,footskip=1cm,top=3.5cm,bottom=2.5cm}
2570   \fi
2571   \ifAMC@watermark
2572     \ifAMC@correthead\else
2573       \def\AMC@note{\begin{minipage}{0.65\linewidth}
2574         \AMC@LR{\textcolor{blue}{\AMC@loc@message}}
2575       \end{minipage}}
2576     }
2577   \fi
2578 \fi
2579 \fi
```


4.26 Initialisation

Initialisation of the check counter:

```
2580 \AMCid@check=\AMCid@checkmax\advance\AMCid@check@one
```

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

```
2581 \ifAMC@ensemble\AMCmessage{VAR:ensemble=1}\fi
2582 \ifAMC@inside@box\AMCmessage{VAR:insidebox=1}\fi
2583 \ifAMC@outside@box\AMCmessage{VAR:outsidebox=1}\fi
2584 \ifAMC@postcorrect\AMCmessage{VAR:postcorrect=1}\fi
```

Preparing writing to .xy file :

```
2585 \ifAMC@calibration
2586 \newwrite\AMC@XYFILE%
2587 \immediate\openout\AMC@XYFILE\jobname.xy%
2588 \immediate\write\AMC@XYFILE{\string\version{\AMC@VERSION}}
2589 \immediate\write\AMC@XYFILE{\string\with{codedigit=\AMCcodeID@mode}}
2590 \immediate\write\AMC@XYFILE{\string\with{version=\AMC@VERSION}}
2591 \immediate\write\AMC@XYFILE{\string\with{ensemble=\ifAMC@ensemble yes\else no\fi}}
2592 \immediate\write\AMC@XYFILE{\string\with{insidebox=\ifAMC@inside@box yes\else no\fi}}
2593 \immediate\write\AMC@XYFILE{\string\with{outsidebox=\ifAMC@outside@box yes\else no\fi}}
2594 \immediate\write\AMC@XYFILE{\string\with{postcorrect=\ifAMC@postcorrect yes\else no\fi}}
2595 \immediate\write\AMC@XYFILE{\string\with{extractonly=\ifAMC@extractOnly yes\else no\fi}}
2596 \immediate\write\AMC@XYFILE{\string\with{lang=\AMC@lang@code}}
2597 \ifx\AMCNombreCopies\undefined%
2598 \immediate\write\AMC@XYFILE{\string\with{ncopies=default}}}%
2599 \else%
2600 \immediate\write\AMC@XYFILE{\string\with{ncopies=\AMCNombreCopies}}}%
2601 \fi%
2602 \fi
```

4.27 French command names

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

```
2603 \let\reponses=\choices\let\endreponses=\endchoices
2604 \let\reponseshoriz=\choiceshoriz\let\endreponseshoriz=\endchoiceshoriz
2605 \let\reponsesperso=\choicescustom\let\endreponsesperso=\endchoicescustom
2606 \let\bonne=\correctchoice
2607 \let\mauvaise=\wrongchoice
2608 \let\bareme=\scoring
2609 \let\baremeDefautM=\scoringDefaultM
2610 \let\baremeDefautS=\scoringDefaultS
2611 \def\exemplaire{\AMC@loc@FR\onecopy}
2612 \@ifpackageloaded{environ}{%
2613   \let\copieexamen=\examcopy\let\endcopieexamen=\endexamcopy}{%
2614 \let\melangegroupe=\shufflegroup
2615 \let\restituegroupe=\insertgroup
2616 \let\alafin=\lastchoices
2617 \let\formulaire=\AMCform
2618 \let\AMCdebutFormulaire=\AMCformBegin
2619 \let\champnom=\namefield
2620 \let\choixIntervalles=\AMCIntervals
```

5 Outputs

In the .xy file, 0/⟨*n*⟩ means student sheet number 0 (there is only one “student sheet” numbered 0 for this document as we did not use \onecopy) and page number ⟨*n*⟩ inside this student sheet. Then, each instance of the \tracepos command shows *x* and *y* positions as arguments #2 and #3 (unit is sp, such that 65536×72.27 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{0/34: __zone:id: __n}{0sp}{27505181sp}{square}
\tracepos{0/34: __zone:id: __n}{5873801sp}{0sp}{square}
\tracepos{0/34: __zone:id: __n}{15861297sp}{0sp}{square}
\tracepos{0/34: __zone:id: __n}{0sp}{24520003sp}{square}
```

5.2 AMCboxedchar command

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

```
\tracepos{0/34:test}{22855914sp}{11385956sp}{square}
\tracepos{0/34:test}{23561334sp}{10680536sp}{square}
```

5.3 AMCcode command

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

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

\tracepos{0/58:case:code[4]:17,3}{27245723sp}{42694064sp}{square}
\boxchar{0/58:case:code[4]:17,3}{2}
\tracepos{0/58:case:code[4]:17,4}{26540303sp}{42285372sp}{square}
\tracepos{0/58:case:code[4]:17,4}{27245723sp}{41579952sp}{square}
\boxchar{0/58:case:code[4]:17,4}{3}
\tracepos{0/58:case:code[4]:17,5}{26540303sp}{41171260sp}{square}
\tracepos{0/58:case:code[4]:17,5}{27245723sp}{40465840sp}{square}
\boxchar{0/58:case:code[4]:17,5}{4}
\tracepos{0/58:case:code[4]:17,6}{26540303sp}{40057148sp}{square}
\tracepos{0/58:case:code[4]:17,6}{27245723sp}{39351728sp}{square}
\boxchar{0/58:case:code[4]:17,6}{5}
\tracepos{0/58:case:code[3]:18,1}{28032160sp}{45627708sp}{square}
\tracepos{0/58:case:code[3]:18,1}{28737580sp}{44922288sp}{square}
\boxchar{0/58:case:code[3]:18,1}{0}
\tracepos{0/58:case:code[3]:18,2}{28032160sp}{44513596sp}{square}
\tracepos{0/58:case:code[3]:18,2}{28737580sp}{43808176sp}{square}
\boxchar{0/58:case:code[3]:18,2}{1}
\tracepos{0/58:case:code[3]:18,3}{28032160sp}{43399484sp}{square}
\tracepos{0/58:case:code[3]:18,3}{28737580sp}{42694064sp}{square}
\boxchar{0/58:case:code[3]:18,3}{2}
\tracepos{0/58:case:code[3]:18,4}{28032160sp}{42285372sp}{square}
\tracepos{0/58:case:code[3]:18,4}{28737580sp}{41579952sp}{square}
\boxchar{0/58:case:code[3]:18,4}{3}
\tracepos{0/58:case:code[3]:18,5}{28032160sp}{41171260sp}{square}
\tracepos{0/58:case:code[3]:18,5}{28737580sp}{40465840sp}{square}
\boxchar{0/58:case:code[3]:18,5}{4}
\tracepos{0/58:case:code[3]:18,6}{28032160sp}{40057148sp}{square}
\tracepos{0/58:case:code[3]:18,6}{28737580sp}{39351728sp}{square}
\boxchar{0/58:case:code[3]:18,6}{5}
\tracepos{0/58:case:code[2]:19,1}{29524017sp}{45627708sp}{square}
\tracepos{0/58:case:code[2]:19,1}{30229437sp}{44922288sp}{square}
\boxchar{0/58:case:code[2]:19,1}{0}
\tracepos{0/58:case:code[2]:19,2}{29524017sp}{44513596sp}{square}
\tracepos{0/58:case:code[2]:19,2}{30229437sp}{43808176sp}{square}
\boxchar{0/58:case:code[2]:19,2}{1}
\tracepos{0/58:case:code[2]:19,3}{29524017sp}{43399484sp}{square}
\tracepos{0/58:case:code[2]:19,3}{30229437sp}{42694064sp}{square}
\boxchar{0/58:case:code[2]:19,3}{2}
\tracepos{0/58:case:code[2]:19,4}{29524017sp}{42285372sp}{square}
\tracepos{0/58:case:code[2]:19,4}{30229437sp}{41579952sp}{square}
\boxchar{0/58:case:code[2]:19,4}{3}
\tracepos{0/58:case:code[2]:19,5}{29524017sp}{41171260sp}{square}
\tracepos{0/58:case:code[2]:19,5}{30229437sp}{40465840sp}{square}
\boxchar{0/58:case:code[2]:19,5}{4}
\tracepos{0/58:case:code[2]:19,6}{29524017sp}{40057148sp}{square}
\tracepos{0/58:case:code[2]:19,6}{30229437sp}{39351728sp}{square}
\boxchar{0/58:case:code[2]:19,6}{5}
\tracepos{0/58:case:code[1]:20,1}{31015874sp}{45627708sp}{square}
\tracepos{0/58:case:code[1]:20,1}{31721294sp}{44922288sp}{square}

```

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

```

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

Table 1: French equivalent commands

Contents

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

4.8.4	Scoring zones	40
4.8.5	Binary boxes	40
4.9	Checking Environment	42
4.10	Handling groups of questions	42
4.11	Questions	46
4.11.1	Managing answers	47
4.11.2	Separate answer sheet	47
4.11.3	Formatting answers	51
4.11.4	Score zones	52
4.11.5	Formatting questions	55
4.11.6	Explanations	57
4.12	Scoring	57
4.13	Numerical data	58
4.13.1	Codes	58
4.13.2	Numerical questions	61
4.13.3	Intervals	75
4.14	Open questions	76
4.15	Boxes with letters only	78
4.16	Page formatting	79
4.16.1	Watermark	79
4.16.2	Signs for scan analysis	80
4.17	Defining a single exam copy content	82
4.18	Pre-association	83
4.19	Package options	83
4.20	Survey add-on	85
4.21	Package Errors	86
4.22	Optional features	86
4.23	Use with recent LuaTeX versions	87
4.24	External control	87
4.25	Page layout	88
4.26	Initialisation	89
4.27	French command names	89
5	Outputs	90
5.1	<code>namefield</code> command	90
5.2	<code>AMCboxedchar</code> command	90
5.3	<code>AMCcode</code> command	90

Index

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

Symbols			1397,	1398,	1418,	1837,	1838,	1839,
\"	148, 154	1419,	1424,	1425,	1840,	1841,	1842,
\@aucune	839, 843, 844	1426,	1427,	1428,	1843,	1844,	1846,
\@firstoftwo	659	1429,	1434,	1436,	1851,	1852,	1855,
\@ifstar	1003, 1004	1437,	1438,	1439,	1856,	1860,	1861,
\@secondoftwo	661	1440,	1442,	1444,	1862,	1864,	1865,
\@skiphyperreffalse	...	1172	1445,	1446,	1447,	1867,	1868,	1869,
\@skiphyperreftrue		1168, 1172	1450,	1457,	1458,	1870,	1871,	1873,
\@tempboxa	.	2217, 2221, 2222	1459,	1461,	1463,	1874,	1877,	1879,
\@tempdimb	2206, 2208	1464,	1466,	1468,	1884,	1885,	1886,
\@tempdimc	2207, 2208	1473,	1474,	1476,	1887,	1889,	1890,
_	1207	1478,	1479,	1480,	1891,	1892,	1893,
\~	189, 190, 195, 196	1482,	1488,	1493,	1894,	1895,	1898,
A			1505,	1539,	1541,	1899,	1900,	1901,
\aa	175	1543,	1545,	1547,	1905,	1907,	1909,
\alafin	21, 2616	1550,	1553,	1556,	1912,	1917,	1918,
\amc	866, 877,	1559,	1562,	1565,	1919,	1920,	1921,
		879, 881, 882, 884,	1568,	1571,	1574,	1922,	1923,	1925,
		887, 888, 890, 893,	1577,	1586,	1588,	1926,	1927,	1928,
		894, 897, 898, 899,	1596,	1600,	1601,	1932,	1933,	1935,
		900, 902, 904, 905,	1602,	1608,	1610,	1941,	1948,	1953,
		907, 909, 910, 912,	1622,	1623,	1624,	1955,	1956,	1962,
		914, 915, 918, 919,	1625,	1626,	1627,	1971,	1973,	1974,
		921, 922, 924, 925,	1628,	1629,	1632,	1975,	1976,	1980,
		927, 928, 930, 931,	1635,	1639,	1642,	1981,	1982,	1986,
		933, 936, 937, 939,	1645,	1649,	1652,	1987,	1988,	1989,
		941, 945, 946, 947,	1656,	1659,	1664,	1991,	1992,	1993,
		948, 1270, 1271, 1272,	1666,	1668,	1675,	2006,	2008,	2014,
		1273, 1274, 1275,	1677,	1680,	1689,	2023,	2026,	2027,
		1276, 1278, 1285,	1690,	1691,	1692,	2033,	2034,	2035,
		1288, 1293, 1297,	1694,	1696,	1699,	2036,	2037,	2038,
		1305, 1308, 1309,	1704,	1705,	1707,	2039,	2041,	2042, 2045
		1310, 1313, 1318,	1711,	1713,	1716,	\AMC@addpages	2329, 2331, 2333	
		1320, 1323, 1324,	1717,	1718,	1719,	\AMC@affecte	279, 1183
		1325, 1326, 1328,	1721,	1722,	1723,	\AMC@amclog	17
		1330, 1332, 1333,	1726,	1729,	1734,	\AMC@answerBox	478
		1335, 1336, 1340,	1736,	1739,	1743,	\AMC@answerBox@	...	359,
		1341, 1342, 1344,	1744,	1745,	1747,		486, 533, 535, 551,	
		1345, 1346, 1348,	1750,	1755,	1761,		553, 619, 621, 629, 631	
		1353, 1355, 1357,	1763,	1765,	1768,	\AMC@binaryBoxes	652
		1365, 1370, 1371,	1776,	1777,	1784,	\AMC@binaryCode	...	602,
		1372, 1374, 1376,	1785,	1787,	1788,		653, 2237, 2238, 2239	
		1379, 1380, 1381,	1789,	1791,	1794,	\AMC@box	520, 1053,
		1382, 1384, 1386,	1795,	1798,	1799,		1054, 1057, 2473, 2474	
		1389, 1390, 1392,	1801,	1807,	1811,	\AMC@boxcolor	377, 503, 2128	
		1393, 1395, 1396,	1812,	1814,	1817,	\AMC@boxcolor@	
			1825,	1835,	1836,		377, 378, 379, 381,	

399, 408, 420, 424, 441	\AMC@loc@catalog 99, 112,	\AMC@mem@add@ifneeded ..
\AMC@boxeddown 391, 491, 502	126, 139, 153, 167, 864, 1213
\AMC@boxedheight	180, 192, 206, 218, 2394	\AMC@mem@addsingle@ifneeded
. 404, 418, 419, 421,	\AMC@loc@corrected	956, 996, 998, 1000, 1002
422, 425, 426, 470, 98, 111,	\AMC@mem@addvar
493, 497, 498, 510, 513	125, 138, 152, 166,	\AMC@mem@aid
\AMC@boxedrule 396,	179, 191, 205, 217, 2233	\AMC@mem@answer
418, 419, 420, 489, 500	\AMC@loc@DE 963, 1026, 1033,
\AMC@boxedwidth	\AMC@loc@draft .. 93, 106,	1034, 1035, 1040, 1046
. 407, 418, 419, 421,	120, 133, 147, 161,	\AMC@mem@category . 925,
422, 425, 426, 470,	174, 187, 200, 213, 2204	959, 2018, 2164, 2200
492, 497, 499, 510, 511	\AMC@loc@ES	\AMC@mem@clear ... 882, 2321
\AMC@catalogMode	\AMC@loc@explain 100, 113,	\AMC@mem@next . 902, 958, 972
..... 2389, 2397, 2556	140, 154, 193, 219, 1244	\AMC@mem@openQuestion ..
\AMC@CBtaille 600, 2261, 2403	\AMC@loc@FR .. 132, 2399, 2611 970, 1212
\AMC@checkedbox	\AMC@loc@IT	\AMC@mem@qidaffname 919, 973
. 359, 380, 383, 433,	\AMC@loc@JA	\AMC@mem@show . 946, 983, 992
435, 445, 451, 461,	\AMC@loc@message 94, 107,	\AMC@mem@show@filter 947, 987
619, 621, 1053, 1613,	121, 134, 148, 162,	\AMC@mn@leftmargin
1633, 1645, 1649, 2473	175, 188, 201, 214, 2574	. 1063, 1071, 1074, 1080
\AMC@chiffres	\AMC@loc@namesurname ...	\AMC@mn@rightmargin
1258	. 104, 117, 130, 144,	. 1064, 1072, 1075, 1078
\AMC@crosschar 402, 505	158, 171, 184, 197, 210	\AMC@mn@sep . 1062, 1078, 1080
\AMC@crossrule 424, 490, 506	\AMC@loc@NL	\AMC@mn@test 1061, 1068, 1077
\AMC@definitnumero . 280, 284	\AMC@loc@NO	\AMC@multi@report 1323, 2327
\AMC@draw@crossfalse .. 382	\AMC@loc@none .. 101, 114,	\AMC@multiclear . 1328, 2318
\AMC@draw@crosstrue ... 384	127, 141, 155, 168,	\AMC@NCBcheck 596, 2239, 2403
\AMC@error@explain	181, 194, 207, 220, 839	\AMC@NCBetud . 596, 2237, 2402
..... 1244, 1246, 2499	\AMC@loc@PT	\AMC@NCBpage . 596, 2238, 2402
\AMC@fillcolor@	186	\AMC@new@savebox
. 380, 398, 399, 415, 420	\AMC@loc@q 360, 2428,
\AMC@fin@rep .. 826, 1011,	124, 137, 151, 165,	2432, 2433, 2434, 2435
1015, 1019, 1022, 1025	178, 190, 204, 216, 1175	\AMC@note
\AMC@formBox	\AMC@loc@qf	2234,
520	96, 109,	2235, 2269, 2279, 2573
\AMC@formBox@	123, 136, 150, 164,	\AMC@numeric@alsocorrect
520	177, 189, 203, 215, 848 1592,
\AMC@fullGroupsfalse ... 34	\AMC@loc@question	1944, 1948, 1958, 1962
\AMC@fullGroupstrue 102, 115,	\AMC@numeric@keepas
..... 2367, 2393	128, 142, 156, 169, 1590, 1937, 1938
\AMC@if@separate@question	182, 195, 208, 221, 227	\AMC@numeric@scoreapprox
..... 877,	\AMC@loc@questions 1582, 1903, 1913
952, 957, 965, 971, 103, 116,	\AMC@numeric@scoreexact
1403, 2018, 2164, 2200	129, 143, 157, 170, 1580, 1902, 1908
\AMC@imax . 713, 725, 726, 727	183, 196, 209, 222, 227	\AMC@numeric@scorewrong
\AMC@intituleHead	\AMC@logchar 1584, 1910, 1913
..... 2233, 2250, 2394	287, 388	\AMC@numerotation
\AMC@keepmemoryfalse .. 2319	\AMC@logfile . 17, 18, 19, 288 279, 283, 284
\AMC@keepmemorytrue ... 993	\AMC@LR	\AMC@outside@sep .. 495, 501
\AMC@keyBox@	21,	\AMC@oval@radius
. 549, 1296, 1615, 1618	390, 2209, 2256, 2257, 420, 494, 511, 513
\AMC@keys@line	2258, 2259, 2260, 2574	\AMC@ovalbox@ 434, 448, 2434
2396	\AMC@makeovalbox	\AMC@ovalbox@F 435, 446, 2435
\AMC@keys@next 413, 432, 433, 434, 435	
2395	\AMC@mem@add	
\AMC@lang@code	907,	
. 2364, 2399, 2400, 2596	953, 960, 967, 974, 1404	
\AMC@loc@CA		
105		

\AMC@ovalbox@R	\AMC@SRbit	239	\AMC@sza@init@margin ..	1133
.... 431, 432, 454, 2432	\AMC@SRconst ..	230, 234, 237	\AMC@sza@init@none	1121
\AMC@ovalbox@RF 433, 452, 2433	\AMC@SRmax	245, 273	\AMC@sza@init@question	1125
\AMC@pagepos	\AMC@SRnextByte	245	\AMC@sza@width	
\AMC@premierecopie	\AMC@SRnum 247, 248, 250,	 1119, 1137, 1142	
.. 601, 2303, 2306, 2404	251, 255, 258, 273, 274		\AMC@tempenv	657, 658
\AMC@prepare ..	\AMC@SRset 231, 243, 244, 246		\AMC@tracebox	
\AMC@prepare@element ...	\AMC@SRtest	240, 251	. 293, 347, 440, 466,	
..... 676, 684, 799	\AMC@SRvalue	242	584, 586, 589, 1042, 2227	
\AMC@printformoutside@false	\AMC@stepQuestion		\AMC@tracechar 335, 537, 555	
..... 524 1165, 1199, 1237		\AMC@tracepos	
\AMC@printformoutside@true	\AMC@sti 260, 268, 272, 275, 276	 294, 405, 411,	
..... 526, 527	\AMC@stil	261,	1028, 1031, 1205, 1213	
\AMC@printkeyoutside@false	269, 270, 271, 273, 277		\AMC@traceposx	303, 323
..... 541	\AMC@surveyfalse	2413	\AMC@traceposy 312, 322, 324	
\AMC@printkeyoutside@true	\AMC@surveytrue	2414	\AMC@unnumero	279
..... 543, 545	\AMC@sz@box	1084,	\AMC@use@box	362,
\AMC@qaff 849, 1165, 1207, 1240	1086, 1088, 1091, 1134		446, 448, 452, 454, 2430	
\AMC@save@box 361, 416, 2429	\AMC@sz@callin		\AMC@VERSION	2588, 2590
\AMC@setcolors@ 1096, 1101, 1109,		\AMC@watertext ..	2204, 2215
.... 376, 397, 414, 439	1111, 1176, 1208, 1209		\AMC@XYFILE	296,
\AMC@shape@form	\AMC@sz@callin@question	1086	305, 314, 326, 331,	
\AMC@shape@form@base 460, 474	\AMC@sz@callout		332, 333, 337, 343,	
\AMC@shape@form@ticked 1095, 1100,		345, 1186, 1320, 2311,	
..... 462, 464, 467	1104, 1106, 1202, 1203		2355, 2360, 2361,	
\AMC@shape@none	\AMC@sz@callout@margin	1088	2586, 2587, 2588,	
\AMC@shape@oval	\AMC@sz@callout@margin	1091	2589, 2590, 2591,	
\AMC@shape@square	\AMC@sz@depth 1084, 1094, 1099		2592, 2593, 2594,	
\AMC@shapename	\AMC@sz@height		2595, 2596, 2598, 2600	
. 293, 301, 310, 319, 496 1084, 1093, 1098		\AMC@addpagesto	2329
\AMC@shapename@ 293, 391, 507	\AMC@sz@init@margin ..	1090	\AMC@answer	
\AMC@shapeprepare	\AMC@sz@width 1084, 1092, 1097		. 1046, 1047, 2117, 2179	
..... 507, 515, 2436	\AMC@sza@box 1119, 1127, 1130		\AMC@association	2353
\AMC@shapeprepare@form 459	\AMC@sza@callin 858, 859,		\AMC@beforeQuestion	
\AMC@shapeprepare@none 476	1141, 1146, 1154, 1156	 1174, 1201, 1280	
\AMC@shapeprepare@oval 430	\AMC@sza@callin@margin	1131	\AMC@beginAnswer	
\AMC@shapeprepare@square 394	\AMC@sza@callin@margin	1135	. 1018, 1047, 2116, 2178	
\AMC@shuffletoks	\AMC@sza@callin@none ..	1123	\AMC@beginQuestion	
.... 260, 746, 749, 819	\AMC@sza@callin@question	 1174, 1207,	
\AMC@smashbox 1127		1220, 1226, 1240, 1279	
368, 369, 370, 371, 374	\AMC@sza@callout 854, 855,		\AMC@bin@begin	651
\AMC@smashboxheight	1140, 1145, 1149, 1151		\AMC@bin@did 606, 614, 617,	
. 365, 368, 369, 370, 373	\AMC@sza@callout@margin	1130	619, 624, 627, 629, 635	
\AMC@smashcentered	\AMC@sza@callout@margin		\AMC@bin@digit	
. 366, 408, 409, 441, 442 1134		. 604, 637, 640, 646, 649	
\AMC@SR	\AMC@sza@callout@none .	1122	\AMC@bin@hsep 608, 611, 615, 625	
231, 234, 235, 236,	\AMC@sza@callout@question		\AMC@bin@id	605,
237, 239, 241, 242, 2312 1126		610, 618, 619, 628, 629	
\AMC@SR@count	\AMC@sza@depth		\AMC@bin@ndigits	
235, 237, 245, 249, 1119, 1139, 1144	 607, 609, 646, 649	
252, 255, 256, 257, 258	\AMC@sza@height		\AMC@bin@number	
\AMC@SR@time 1119, 1138, 1143		. 603, 636, 639, 641, 643	
246, 2380	\AMC@sza@init@margin ..	1129	\AMC@bin@one	613, 641
\AMC@SR@advance 232, 239, 240				

\AMCbin@sequence .. 602,	\AMCform 16, 864, 2617	1183, 1186, 1213,
636, 641, 642, 647, 648	\AMCformAfterQuestion ..	1296, 1615, 1618, 2060
\AMCbin@style 612, 636 847, 1213	\AMCid@quest 26, 331, 332,
\AMCbin@zero .. 623, 642, 647	\AMCformAnswer 846	333, 565, 568, 570,
\AMCbloc 1181	\AMCformAnswerA ... 863, 967	574, 576, 584, 586,
\AMCbo@help . 2168, 2169, 2182	\AMCformatChoices	589, 915, 973, 1028,
\AMCboHide 2195, 2201	. 1401, 2019, 2165, 2201	1031, 1042, 1183,
\AMCboOpts 2172	\AMCformBeforeQuestion .	1185, 1186, 1205,
\AMCboShow 2173, 2201 846, 852	1213, 1296, 1616,
\AMCbotextGoto .. 2167, 2196	\AMCformBegin . 16, 864, 2618	1619, 1848, 2099, 2175
\AMCboxColor 518, 2261	\AMCformFilter 985	\AMCIDBoxesA 2237, 2241
\AMCboxDimensions . 519, 636	\AMCformHSpace 66, 862, 2179	\AMCIDBoxesABC .. 2240, 2263
\AMCBoxedAnswers 1005, 2371	\AMCformQuestion 846	\AMCIDBoxesB 2238, 2242
\AMCBoxOnly 2167	\AMCformQuestionA . 850, 974	\AMCIDBoxesC 2239, 2242
\AMCboxOutsideLetter ...	\AMCformQuestionN . 849, 857	\AMCids@height . 77, 88, 2262
..... 520, 2118, 2180	\AMCformS 864	\AMCids@sidefalse ... 80, 82
\AMCboxStyle	\AMCformVSpace 66, 846	\AMCids@sidetrue 84
..... 20, 489, 2371, 2421	\AMCgroup@pre 742, 765	\AMCids@topfalse 80, 84
\AMCccircle 2223, 2226	\AMCgrouploop@next	\AMCids@toptrue 82
\AMCchoiceLabel ... 478, 559 767, 780, 798	\AMCids@width .. 76, 87, 2265
\AMCchoiceLabelFormat ..	\AMCgrouploop@prep	\AMCidsPosition 74
..... 392, 478 753, 778, 796	\AMCidsVar 78
\AMCcleardoublepage 16, 2338	\AMCgrouppre@cyclic ... 736	\AMCidsVarN 78, 79
\AMCcode 1397	\AMCgrouppre@fixed 698	\AMCif@env .. 656, 1244, 1246
\AMCcodeGrid 15, 1258	\AMCgrouppre@withoutreplacement	\AMCifcategory 948
\AMCcodeGridInt ... 15, 1258 714	\AMCinterBquest ... 71, 1213
\AMCcodeH 1398	\AMCgrouppre@withreplacement	\AMCinterBrep 66, 1032
\AMCcodeHspace 705	\AMCinterIquest ... 70, 1213
..... 1259, 1300, 1359	\AMChorizAnswerSep	\AMCinterIrep 66, 1009
\AMCcodeID 1264, 1339 1036, 1037, 1045	\AMCIntervalFormat 2030, 2041
\AMCcodeID@@	\AMChorizBoxSep	\AMCIntervals 17, 2030, 2620
. 1263, 1267, 2412, 2562	. 1038, 1039, 1042, 1044	\AMClabel 2346, 2351
\AMCcodeID@dot 1262	\AMCid@check	\AMCload@reponse . 830,
\AMCcodeID@mode . 1265, 2589	27, 328, 2239, 2244,	832, 1053, 1057, 2473
\AMCcodeID@squarebrackets	2245, 2262, 2266, 2580	\AMCload@counter
..... 1261	\AMCid@checkmax 25, 808, 809, 810,
\AMCcodeVspace 596, 2245, 2402, 2580	811, 814, 815, 819,
. 1260, 1298, 1358, 1361	\AMCid@etud 28, 290,	822, 823, 824, 825, 828
\AMCcompleteMulti .. 12, 63	298, 307, 316, 328,	\AMCload@reponse .. 813, 832
\AMCcurrentenv 655, 658, 1198	331, 332, 333, 338,	\AMClocalized 92
\AMCdebutFormulaire 21, 2618	353, 474, 2236, 2237,	\AMCloop@k 719, 725,
\AMCdecimalPoint	2262, 2266, 2274,	729, 731, 752, 760, 769
..... 1527, 1708, 1740	2302, 2303, 2305,	\AMCmarginNote
\AMCdefault@groupmode ..	2312, 2317, 2324, 1065, 1091, 1134
..... 673, 687, 688	2325, 2345, 2346, 2355	\AMCmem@elt@cat 922, 934, 938
\AMCdontAnnotate . 331, 2133	\AMCid@etudfin 30,	\AMCmessage . 17, 280, 592,
\AMCdontScan 331, 2131	2304, 2305, 2306, 2324	992, 1051, 1056, 1185,
\AMCdum@reponses . 819, 833	\AMCid@etudstart	1192, 1193, 1194,
\AMCemptybox 1059, 1084, 1119 29, 2302, 2325	1213, 1217, 1254,
\AMCendAnswer	\AMCid@name . 65, 565, 568,	1255, 1256, 1257,
. 1019, 1047, 2116, 2178	570, 574, 576, 916,	1295, 1409, 1612,
\AMCexponent 1529, 1978, 1990	973, 1028, 1031, 1042,	1644, 1646, 1651,
\AMCexternalQuestion .. 1189		1653, 1658, 1660,

1669, 1682, 1698,	\AMCocol@FrameRule	1296, 1402, 1404,
1790, 1793, 1826, 2052, 2076, 2149	1611, 1612, 1616,
1828, 1848, 1899,	\AMCOpen 2048	1619, 2470, 2471, 2474
1906, 1923, 1925,	\AMCopen@answer	\AMCrep@fini 830, 833, 837, 845
1928, 1930, 1933,	. 2054, 2061, 2090, 2106	\AMCrep@init . . 826, 1010,
1935, 1938, 1942,	\AMCopen@boxframerule . .	1013, 1017, 1021, 1024
1945, 1947, 1954, 2073, 2074, 2121	\AMCrep@itemize . 1007, 1026
1959, 1961, 2001,	\AMCopen@boxmargin	\AMCrep@ligne . . . 1013, 1026
2099, 2156, 2175, 2071, 2072, 2120	\AMCrep@o 827, 829
2192, 2300, 2317,	\AMCopen@contentcommand	\AMCrep@perso . . . 1017, 1026
2420, 2424, 2471, 2062, 2150	\AMCrep@r 831
2581, 2582, 2583, 2584	\AMCopen@framerule	\AMCrep@tikz 1021, 1026
\AMCncol@Background 2077, 2078, 2148	\AMCrep@tikzmat . 1024, 1026
. 1537, 1970	\AMCopen@Hspace . . 2063,	\AMCretick 331, 2132
\AMCncol@Border . 1533, 1970	2064, 2119, 2124, 2136	\AMCrien@deux 813, 830
\AMCncol@BorderWidth . . .	\AMCopen@LineHeight	\AMCscoreZone 580, 1059
. 1535, 1969 2067, 2068, 2089	\AMCscoreZoneAnswerSheet
\AMCneeds@storeboxfalse 2409	\AMCopen@Lines 1059
\AMCneeds@storeboxtrue 2410 2069, 2070, 2094	\AMCsection 995
\AMCnobloc 1180, 1284	\AMCopen@lines 2087	\AMCsectionNumbered 995, 1003
\AMCnoCompleteMulti	\AMCopen@lineuptext 2056,	\AMCsectionStar . . 999, 1003
. 12, 64, 2115	2079, 2104, 2107, 2110	\AMCsetFoot 2235
\AMCNombreCopies	\AMCopen@question	\AMCsetScoreZone
. 2298, 2299, 2597, 2600	. 2055, 2060, 2123, 2124 1117, 1118, 1164
\AMCnoScoreZone . 1164, 1281	\AMCopen@Width	\AMCsetScoreZoneAnswerSheet
\AMCnntextGoto . . 1518, 2010 2065, 2066, 2088 1162, 1163, 1164
\AMCnntextSign 1525, 1759, 1760	\AMCopenHide 2159, 2165	\AMCshowSignificantDigits
\AMCnntextVHead . . 1519, 1712	\AMCopenOpts 2086 1466
\AMCnum@copies . 31, 2298,	\AMCopenShow 2097, 2165	\AMCsignificantDigits . 1461
2299, 2300, 2301, 2304	\AMCotextGoto . . . 2048, 2160	\AMCstartWithQuestion . .
\AMCnum@Questions . 820, 822	\AMCotextReserved 2049, 1182, 1190, 1200
\AMCnumeric@Hspace	2059, 2135, 2139, 2141	\AMCstudentlabel
. . 1520, 1532, 1670,	\AMCoutsideLabelFormat 520	. 2346, 2347, 2348, 2349
1672, 1708, 1727, 1990	\AMCpageref 2349	\AMCStudentNumber . . . 2236
\AMCnumeric@Vspace	\AMCpostNquest 72, 1999	\AMCstudentslistfile . . 2358
. 1521, 1531,	\AMCpostOquest 73, 2152	\AMCsubjectPageTag 2273, 2278
1643, 1650, 1657,	\AMCqlabel 2350	\AMCsubsection 995
1684, 1700, 1714,	\AMCquestionaff 1165	\AMCsubsectionNumbered .
1748, 1760, 1977, 1979	\AMCquestionNumberfalse 1282 997, 1004
\AMCnumericChoices . 18, 1401	\AMCquestionNumbertrue . 45	\AMCsubsectionStar 1001, 1004
\AMCnumericChoicesPlain	\AMCrandomseed 243	\AMCsw@p 260
. 2017, 2024	\AMCref 2346, 2346	\AMCsw@p@ 262, 264, 266
\AMCnumericHide . 2014, 2019	\AMCrep@@count 834, 836, 838	\AMCsz@loggedfalse . . . 580
\AMCnumericOpts 1608	\AMCrep@bloc 1008, 1026	\AMCsz@loggedtrue 593
\AMCnumericShow . 2006, 2019	\AMCrep@count . . 565, 568,	\AMCtext 91
\AMCnumero 1166, 2316	570, 574, 576, 807,	\AMCtok@ik
\AMCocol@Background	816, 836, 838, 842,	. 719, 720, 722, 727,
. 2050, 2057, 2122	928, 966, 1007, 1013,	728, 729, 730, 731,
\AMCocol@BoxFrameRule . .	1017, 1021, 1024,	751, 759, 772, 781, 800
. 2051, 2075, 2122	1028, 1031, 1042,	\AMCtok@k 664,
\AMCocol@Foreground	1050, 1051, 1054,	679, 680, 681, 685, 800
. 2053, 2058,	1055, 1056, 1057,	\AMCtok@max 665
2127, 2128, 2138, 2141	1290, 1294, 1295,	

\AMCtok@size	666, 754, 755, 756, 757, 765, 773, 782, 801
\AMCw@termark ...	2203, 2247
\AMCw@terprint ..	2203, 2247
amcxyfile (env.)	342
\AMCzone	347, 348
\answer	289, 1050, 2438
answers (option)	10
asbox (option)	10
\association	2355
auto (env.)	2438
automarks (option)	11
B	
\bareme	21, 2608
\baremeDefaultM	21, 2609
\baremeDefaultS	21, 2610
\bf	1528
bloc (option)	21
\bonne	21, 2606
\bool 940, 1273, 1274, 1275, 1297, 1308, 1309, 1332, 1335, 1339, 1346, 1348, 1357, 1601, 1666, 1677, 1692, 1711, 1716, 1757, 1761, 1776, 1789, 1795, 1799, 1814, 1824, 1825, 1851, 1855, 1864, 1905, 1941, 1953, 1971	
box (option)	10
\boxchar	338
\boxput	6, 401, 440
C	
\c	945
calibration (option) ..	5, 10
catalog (option)	10
\CatalogExterne	2554
\champnom	21, 2619
\char	1421
\CheckBox	467
chiffres (option)	21
\choices	2603
choices (env.)	12, 1005
\choicescustom	2605
choicescustom (env.) ..	12, 1005
\choiceshoriz	2604
choiceshoriz (env.) ..	12, 1005
\choixIntervalles ..	21, 2620
\cleargroup	14, 788
\clist ..	1270, 1276, 1285, 1320, 1333, 1337, 1371, 1380, 1381, 1475, 1478, 1489, 1490, 1705, 1706, 1736, 1737, 1837, 1838
\codeDigitExterne ..	2561, 2562
completemulti (option) ..	10
\coordinate	1067
\copieexamen	2613
copieexamen (env.)	21
\copygroup	14, 788
\copygroupfrom	14, 788
correc (option)	21
correcindiv (option)	21
\correctchoice	13, 844, 1050, 2040, 2606
\CorrigeExterne	2538
\CorrigeIndivExterne ..	2538
\crefalias	226
\crefname	227
\cs 877, 881, 882, 884, 887, 890, 893, 897, 902, 904, 907, 909, 912, 914, 918, 919, 921, 924, 925, 927, 930, 931, 933, 936, 945, 946, 947, 948, 1278, 1288, 1293, 1318, 1323, 1325, 1328, 1330, 1370, 1374, 1379, 1384, 1389, 1392, 1395, 1396, 1397, 1398, 1416, 1424, 1434, 1437, 1442, 1446, 1458, 1461, 1463, 1466, 1468, 1474, 1488, 1493, 1505, 1596, 1600, 1608, 1610, 1624, 1639, 1664, 1675, 1690, 1704, 1734, 1755, 1784, 1846, 2006, 2008, 2014, 2023, 2026, 2027, 2035, 2045, 2520, 2521, 2523, 2524, 2526, 2527, 2530, 2531, 2533, 2534	
\csvreader	2445, 2450, 2456, 2465
D	
\define@boolkey	504, 2080, 2081, 2082, 2083, 2084, 2170
\define@choicekey	78, 496, 1102, 1147
digits (option)	10
\ding	468
\dontannotate	332
\dontscan	331
\dotfill	356, 2092, 2108
\draw ...	420, 424, 1078, 1080
E	
\element	14, 664
\endchoices	2603
\endchoicescustom	2605
\endchoiceshoriz	2604
\endcopieexamen	2613
\endexamcopy	2613
\endreponses	2603
\endreponseshoriz	2604
\endreponsesperso	2605
ensemble (option)	21
environments:	
amcxyfile	342
auto	2438
choices	12, 1005
choicescustom ..	12, 1005
choiceshoriz ...	12, 1005
copieexamen	21
examcopy	2510
question	11, 1179
question-auto	2438
questionmult ...	11, 1179
Questionnaires	2438
questionouverte ...	1179
reponses	21
reponseshoriz	21
reponsesperso	21
tikz-multi	1005
tikz-single	1005
values	2438
values-auto	2438
variable-auto	2438
variable-multi	1179
variable-single ...	1179
\evensidemargin ..	1072, 1074
\examcopy	2613
examcopy (env.)	2510
\exemplaire	21, 2611
\exemplairepair	2345
\explain	1242, 2500
\explaincontext	1249

<code>\ExplSyntaxOff</code>	<code>\ifAMC@correc</code>	<code>\ifAMC@watermark</code>
. . . 950, 1400, 1517,	38, 379, 1053, 1250, 51, 2247, 2571
2016, 2029, 2047, 2537	1614, 2090, 2106, 2473	<code>\ifAMC@zoneformulaire</code> ..
<code>\ExplSyntaxOn</code>	<code>\ifAMC@correchead</code> 57, 527, 564, 868
. . . 864, 1268, 1414, 35, 1243, 2247,	<code>\ifAMCcomplete@multi</code> 44, 841
1522, 2021, 2031, 2518	2290, 2306, 2566, 2572	<code>\ifAMCformulaire@dedans</code>
F	<code>\ifAMC@draw@cross</code> 56, 526,
<code>\fancypagestyle</code> 363, 402, 423	567, 583, 1847, 2000,
. 2249, 2254, 2276, 2283	<code>\ifAMC@ensemble</code>	2098, 2155, 2174, 2191
<code>\fbox</code> 2274 54, 525, 542,	<code>\ifAMCids@side</code> 75, 2265
<code>\footrulewidth</code>	563, 582, 867, 979,	<code>\ifAMCids@top</code> 74, 2262
. 2252, 2271, 2281, 2286	982, 986, 991, 1207,	<code>\ifAMCneeds@storebox</code> . . .
<code>\foreach</code> 2464	1407, 1847, 1999, 2409, 2416, 2427
<code>\formulaire</code> 21, 2617	2000, 2098, 2103,	<code>\ifAMCquestionNumber</code> 45, 1172
<code>\fp</code> 1428, 1436,	2117, 2123, 2155,	<code>\ifAMCsz@logged</code> . . . 580, 591
1438, 1440, 1444,	2174, 2179, 2182,	<code>\ifAMCtype@multi</code> . . . 50, 841
1449, 1451, 1453,	2191, 2315, 2581, 2591	<code>\ifAMCune@bonne</code> 49, 843
1457, 1464, 1469,	<code>\ifAMC@extractOnly</code>	<code>\ifcase</code> 79
1476, 1494, 1497, 40, 1967, 2595	<code>\ifcsname</code> . . . 1103, 1108,
1835, 1836, 1860,	<code>\ifAMC@fullGroups</code> . . 34, 755	1113, 1148, 1153, 1158
1861, 1870, 1898,	<code>\ifAMC@inside@box</code>	<code>\ifcsvstrcmp</code>
1899, 1925, 2033, 52, 545, 573,	. 2451, 2457, 2458, 2466
2034, 2036, 2037,	2118, 2180, 2582, 2592	<code>\ifdim</code> 1077
2038, 2039, 2041, 2042	<code>\ifAMC@inside@digit</code> . 55, 479	<code>\ifKV@AMCBoxOnly@ordered</code>
francais (option) 10	<code>\ifAMC@invisible</code> . . . 61, 293 2183
fullgroups (option) 11	<code>\ifAMC@keepmemory</code> 989, 2321	<code>\ifKV@AMCdim@cross</code>
G	<code>\ifAMC@keyline</code> 37, 1206, 1207 383, 398, 415
<code>\group</code> 1420, 1432	<code>\ifAMC@ordre</code> 32, 827	<code>\ifKV@AMCOpen@annotate</code> 2133
H	<code>\ifAMC@outside@box</code>	<code>\ifKV@AMCOpen@dots</code> 2091
<code>\he@dbas</code> . . . 2231, 2256, 2257	53, 525, 543, 2583, 2593	<code>\ifKV@AMCOpen@lineup</code> . . .
<code>\he@dhaut</code> 2232, 2260	<code>\ifAMC@pagelayout</code> 2102, 2145
<code>\he@dtaille</code> . 2228, 2231, 2232	. . . 58, 2244, 2289, 2565	<code>\ifKV@AMCOpen@retick</code> . . 2132
<code>\hfuzz</code> 370	<code>\ifAMC@pdfform</code>	<code>\ifKV@AMCOpen@scan</code> 2131
<code>\ht</code> 368, 1060	62, 351, 2309, 2326, 2419	indivanswers (option) . . 6, 10
I	<code>\ifAMC@plain</code> 48, 2502	init (option) 10
<code>\ifAMC@affichekeys</code>	<code>\ifAMC@postcorrect</code>	<code>\InputIfFileExists</code> 15
. 36, 1206, 1207 59, 844, 2584, 2594	<code>\insertgroup</code> . . . 14, 745, 2615
<code>\ifAMC@asqbloc</code> . 42, 847, 853	<code>\ifAMC@printformoutside</code>	<code>\insertgroupfrom</code> . . . 14, 745
<code>\ifAMC@automarks</code> 523, 532	insidebox (option) 10
60, 979, 2315, 2334, 2341	<code>\ifAMC@printformoutside@</code>	<code>\int</code> 879, 881, 885, 891, 898,
<code>\ifAMC@calibration</code> 522, 529	937, 1272, 1285, 1324,
. 46, 295,	<code>\ifAMC@printkeyoutside</code> .	1326, 1335, 1341,
304, 313, 326, 331, 540, 550	1342, 1365, 1429,
332, 333, 336, 1051,	<code>\ifAMC@printkeyoutside@</code>	1445, 1473, 1476,
1056, 1184, 1191, 539, 547	1478, 1479, 1480,
1216, 1254, 1255,	<code>\ifAMC@qbloc</code> . . . 41, 1174,	1482, 1495, 1498,
1256, 1257, 1295,	1205, 1213, 1239,	1500, 1506, 1507,
1319, 2311, 2317,	1241, 2011, 2161, 2197	1509, 1510, 1512,
2354, 2359, 2471, 2585	<code>\ifAMC@rbloc</code> . 43, 1008, 1011	1602, 1622, 1625,
<code>\ifAMC@catalog</code> 47, 387	<code>\ifAMC@shuffleG</code> 33, 746, 749	1627, 1628, 1629,
	<code>\ifAMC@survey</code> . . . 2413, 2438	1631, 1640, 1665,
	<code>\ifAMC@textPos</code> . 39, 1028,	1676, 1683, 1689,
	1031, 1041, 1205, 1213	1691, 1693, 1699,

1705, 1707, 1712, 1718, 1722, 1726, 1729, 1736, 1738, 1744, 1747, 1750, 1762, 1787, 1788, 1791, 1794, 1798, 1801, 1807, 1811, 1812, 1817, 1818, 1839, 1840, 1841, 1844, 1852, 1855, 1856, 1862, 1867, 1869, 1873, 1886, 1887, 1892, 1900, 1901, 1907, 1909, 1912, 1920, 1927, 1928, 1932, 1933, 1935, 1948, 1955, 1956, 1962, 1976, 1989		\NoWatermarkExterne ... 2538	postcorrect (option) 11 \prg 866, 869, 871, 874, 1381, 1477, 1490
O			
		\oddsidemargin .. 1071, 1075	
		\onecopy 2297, 2440, 2514, 2611	
		options:	Q
	answers 10	\question 1179, 1186	question (env.) 11, 1179
	asbox 10	question-auto (env.) ... 2438	\QuestionIndicative 13, 1254, 1289
	automarks 11	questionmult (env.) . 11, 1179	Questionnaires (env.) .. 2438
	bloc 21	questionouverte (env.) . 1179	
	box 10		R
	calibration 5, 10		\raisebox 1528
	catalog 10		\refstepcounter 1172
	chiffres 21		\reponses 2603
	completemulti 10		reponses (env.) 21
	correc 21		\reponseshoriz 2604
	correcindiv 21		reponseshoriz (env.) 21
	digits 10		\reponsesperso 2605
	ensemble 21		reponsesperso (env.) 21
	francais 10		\restituegroupe ... 21, 2615
	fullgroups 11		\retick 333
	indivanswers 6, 10		\rngstate 2312
	init 10		
	insidebox 10		S
	keys 10		\savebox 361
	nopage 6, 11		\savepos 2521
	noshuffle 10		\sbox 1060
	noshufflegroups 10		\scoring 13, 1221, 1227, 1254, 2608
	nowatermark 4, 10		\scoringDefaultM 13, 1254, 2609
	ordre 21		\scoringDefaultS 13, 1254, 2610
	outsidebox 10		\ScoringExterne 2538
	pdfform 11		\section 996, 1000
	plain 11		separateanswersheet (op- tion) 5, 10, 16
	postcorrect 11		\seq 1271, 1345, 1353, 1419, 1427, 1428, 1429
	separateanswersheet 5, 10, 16		\setdefaultgroupmode .. 687
	storebox 11		\setgroupmode 673, 687
	\or 81, 83		\shuffle@it 819, 833
	ordre (option) 21		\shufflegroup 14, 711, 723, 745, 2614
	outsidebox (option) 10		\shufflegroupslice 728, 731, 748
	\ouverte@vs 1179		\smash 2090, 2106
			\space 2500
			\storebox 2429
			storebox (option) 11
P			
	\PackageError 649, 691, 762, 2499		
	\pageref 2349		
	pdfform (option) 11		
	\pdfpageheight 330, 2533, 2534		
	\pdfpagewidth 330, 2530, 2531		
	\pgfextractx 1068		
	\pgfpointanchor 1069		
	\pgfpointdiff 1068		
	\pgfpointorigin 1068		
	plain (option) 11		
K			
\keys 1304, 1331, 1524, 1597, 1850, 2009			
keys (option) 10			
L			
\lastchoices 826, 2616			
\lastxpos 2524			
\lastypos 2527			
\linebreak 2113			
M			
\m@rque 2227, 2256, 2257, 2258, 2259			
\m@rqueCalage ... 2226, 2227			
\marginpar 1088, 1130			
\mauvaise 21, 2607			
\melangegroupe 21, 2614			
\MessageBreak 2500			
\multiSymbole 1174, 1215, 1233			
N			
\namefield ... 15, 348, 2619			
\namefielddots 349			
\newbox 364			
\newline 1206			
\newsavebox 360			
\newstorebox 2428			
\nobreak 521, 2146			
\node 1033, 1034, 1229			
nopage (option) 6, 11			
noshuffle (option) 10			
noshufflegroups (option) . 10			
\nouveau groupe 664, 789			
nowatermark (option) .. 4, 10			

