

The automultiplechoice package*

Alexis Bienvenue
paamc@passoire.fr

April 17, 2025

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:ec7413fa` from AMC 1.7.0

```

        \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 `\nocancelmark` option.

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

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

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

Question 2 What is the capital of Egypt?

☐ Cayenne
☐ Caracas
☐ Cuito
☐ Conakry
☐ Chisinau

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

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

Question 4 What is the capital of Ghana?

☐ Accra ☐ Addis Ababa ☐ Ankara ☐ Apia

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


2/1/60

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Ireland?

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

Question 3 What is the capital of Ghana?

☐ Apia ☐ Accra ☐ Addis Ababa ☐ Ankara

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

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

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


3/1/60

AMC SAMPLE TEST

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

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

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

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

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

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

Question 3 What is the capital of Ireland?

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

Question 4 What is the capital of Thailand?

☐ Beijing ☐ Bangkok ☐ Bangkok ☐ Beirut ☐ Berlin

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


4/1/60

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Egypt?

☐ Caracas
☐ Cayenne
☐ Cuito
☐ Conakry
☐ Chisinau

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

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

Question 4 What is the capital of Ireland?

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

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

First pages from L^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 Cmaury
☐ E Chisoum

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

Question 2 What is the capital of Egypt?

☐ Cayenne
☐ Caracas
☒ Cairo
☐ 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-su becomes emperor of China

Question 4 What is the capital of Ghana?

☒ Accra
☐ Addis Ababa
☐ Ankara
☐ Apia

1

AMC

SAMPLE TEST

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

- `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
☐ Dakar

Question 3 What is the capital of Ghana?

☐ Apia
☒ Accra
☐ Addis Ababa
☐ Ankara

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

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

1

AMC

SAMPLE TEST

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

- `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 Salyut 1
☒ Death of Louis Armstrong

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

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

Question 3 What is the capital of Ireland?

☐ Dhaka
☐ Dili
☐ Dhaka
☒ 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 Salyut 1
☒ Apollo 14 lands on the Moon
☒ Death of Louis Armstrong
☐ The first commercial Concorde flight takes off

Question 2 What is the capital of Egypt?

☐ Caracas
☐ Cayenne
☒ Cairo
☐ 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-su becomes emperor of China

Question 4 What is the capital of Ireland?

☐ Djibouti
☐ Dili
☐ Dhaka
☐ Dakar
☒ 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.

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

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

As an example again without questions:

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

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

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

```

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

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

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

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

```

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

3.5 Students identification

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

`\AMCcodeGrid`
`\AMCcodeGridInt`

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

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

Numeric choices

Question 8

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

3.8 French command names

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

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>\baremeDefaultM</code>
command	<code>\scoringDefaultS</code>	<code>\baremeDefaultS</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

3.9 Customisation

3.9.1 Boxes

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

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

width for the width of the boxes.

height for the height of the boxes.

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

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

rule for the rule width.

outsidesep for the distance between the box and the letter when printed outside the box.

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

Default values are

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

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

Boxes styling	
Question 9 $2 + 2 =$ Ⓐ 1 Ⓑ 4 Ⓒ 10	<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...

The <code>choicescustom</code> environment	
Question 10 $2+2=$ $\left(\begin{array}{ccc} \square & 4 & \square & 3 & \square & 2 \end{array} \right)$	
<pre>\begin{question}{add} \def\AMCbeginAnswer{\$\Big(\$} \def\AMCendAnswer{\$\Big)\$} \def\AMCanswer#1#2{#1 #2\hfill} 2+2= \begin{choicescustom} \correctchoice{4} \wrongchoice{2} \wrongchoice{3} \end{choicescustom} \end{question}</pre>	

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 \@ifl@t@r\fmtversion{2020/10/01}
4 {}
5 {\RequirePackage{atbegshi}} % \AtBeginShipout
6 \RequirePackage{xkeyval} % \setkeys
7 \RequirePackage{rotating} % \rotatebox
8 \RequirePackage{fancybox} % \boxput
9 \RequirePackage{expl3}
10 \RequirePackage{csvsimple}
11 \RequirePackage{environ}
12 % \end{macrocode}
13 %
14 % First, we read the options that can be given by AMC through the
15 % |jobname-config.tex| file:
16 % \begin{macrocode}
17 \InputIfFileExists{\jobname-config.tex}%
18 {\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.

```
19 \newwrite\AMC@logfile
20 \immediate\openout\AMC@logfile=\jobname.amc
21 \def\AMC@amclog#1{\immediate\write\AMC@logfile{#1}}
22 \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.

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

4.1 Variables

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

The package defines the following counters:

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

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

`\AMCid@etud` current student sheet number.

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

`\AMCid@check` current page checking number.

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

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

It also defines the following switches:

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

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

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

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

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

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

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

`\ifAMC@qbloc` if questions are to be included in \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.

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

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

`\AMCid@name` The package also defines command `\AMCid@name` to be the current question identifier key.
67 `\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
`\AMCinterIrep` sheet.
`\AMCinterBrep` `\AMCformHSpace` is the amount of horizontal space between two answers boxes in a separate
answer sheet.
`\AMCinterIrep` is the amount of vertical space to be added between two answers.
`\AMCinterBrep` is the amount of vertical space between two boxed answers (see `\AMCBoxedAnswers`
and `\ifAMC@rbloc`).
`\AMCinterIquest` is the amount of vertical space left after a question, in standard mode (with-
out 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.
68 `\newdimen\AMCformVSpace\AMCformVSpace=1.2ex`
69 `\newdimen\AMCformHSpace\AMCformHSpace=.3em`
70 `\newdimen\AMCinterIrep\AMCinterIrep=\z@`
71 `\newdimen\AMCinterBrep\AMCinterBrep=.5ex`
72 `\newdimen\AMCinterIquest\AMCinterIquest=\z@`
73 `\newdimen\AMCinterBquest\AMCinterBquest=3ex`
74 `\newdimen\AMCpostNquest\AMCpostNquest=1.5ex`
75 `\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).

```

76 \newif\ifAMCids@top
77 \newif\ifAMCids@side
78 \newdimen\AMCids@width
79 \newdimen\AMCids@height
80 \define@choicekey*{AMCids}{pos}{\AMCidsVar\AMCidsVarN}{none,top,side}{%
81   \ifcase\AMCidsVarN\relax
82     \AMCids@topfalse\AMCids@sidefalse
83   \or
84     \AMCids@toptrue\AMCids@sidefalse
85   \or

```

```

86 \AMCids@topfalse\AMCids@sidetrue
87 \fi
88 }
89 \define@key{AMCids}{width}{\AMCids@width=#1}
90 \define@key{AMCids}{height}{\AMCids@height=#1}
91 \def\AMCidsPosition#1{\setkeys{AMCids}{#1}}
92 \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.

```

93 \def\AMCtext#1#2{\expandafter\def\csname AMC@loc@#1\endcsname{#2}}
94 \def\AMClocalized#1{\csname AMC@loc@#1\endcsname}

```

4.4.1 English

Text indicating draft exams:

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

Message at page bottom when compiled out of AMC gui:

```

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

```

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

```
98 \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):

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

Headers for corrected version and catalog:

```

100 \def\AMC@loc@corrected{Corrected}
101 \def\AMC@loc@catalog{Catalog}

```

Localization text for Explanation

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

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

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

Word for 'question', singular and plural forms:

```

104 \def\AMC@loc@question{question}
105 \def\AMC@loc@questions{questions}

```

Default text to write in the students' name box:

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

4.4.2 Catalan

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

```
107 \def\AMC@loc@CA{
108   \def\AMC@loc@draft{PROJECTE}
109   \def\AMC@loc@message{Pel vostre examen, imprimiu preferiblement
110     els documents compilats amb l'ajuda de auto-multiple-choice.}
111   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
112   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
113   \def\AMC@loc@corrected{Correcci\'}o}
114   \def\AMC@loc@catalog{Cat\'aleg}
115   \def\AMC@loc@explain{\textit{\textbf{Explicaci\'}o : }}}
116   \def\AMC@loc@none{Cap de les respostes \'es correcte.}
117   \def\AMC@loc@question{pregunta}
118   \def\AMC@loc@questions{preguntes}
119   \def\AMC@loc@namesurname{Nom i cognoms:}
120 }
```

4.4.3 Dutch

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

```
121 \def\AMC@loc@NL{
122   \def\AMC@loc@draft{Ontwerp}
123   \def\AMC@loc@message{Gebruik bij uw proefwerk bij voorkeur die
124     documenten welke door auto-multiple-choice zijn aangemaakt.}
125   \def\AMC@loc@qf##1{\textbf{Vraag ##1 :}}
126   \def\AMC@loc@q##1##2{\textbf{Vraag ##1} ##2}
127   \def\AMC@loc@corrected{Correctie}
128   \def\AMC@loc@catalog{Catalogus}
129   \def\AMC@loc@none{Geen van de antwoorden is juist.}
130   \def\AMC@loc@question{vraag}
131   \def\AMC@loc@questions{vragen}
132   \def\AMC@loc@namesurname{Achternaam en voornaam:}
133 }
```

4.4.4 French

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

```
134 \def\AMC@loc@FR{
135   \def\AMC@loc@draft{PROJET}
136   \def\AMC@loc@message{Pour votre examen, imprimez de pr\'ef\'erence
137     les documents compil\'es \'a l'aide de auto-multiple-choice.}
138   \def\AMC@loc@qf##1{\textbf{Question ##1 :}}
139   \def\AMC@loc@q##1##2{\textbf{Question ##1} ##2}
140   \def\AMC@loc@corrected{Correction}
141   \def\AMC@loc@catalog{Catalogue}
142   \def\AMC@loc@explain{\textit{\textbf{Explication : }}}
143   \def\AMC@loc@none{Aucune de ces r\'eponses n'est correcte.}
144   \def\AMC@loc@question{question}
145   \def\AMC@loc@questions{questions}
146   \def\AMC@loc@namesurname{Nom et pr\'enom :}
147 }
```

4.4.5 German

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

```
148 \def\AMC@loc@DE{
149   \def\AMC@loc@draft{ENTWURF}
150   \def\AMC@loc@message{Benutzen Sie f\"ur Ihre Pr\"ufung bevorzugt Dokumente die mit
151     auto-multiple-choice erstellt wurden.}
152   \def\AMC@loc@qf##1{\textbf{Frage ##1 :}}
153   \def\AMC@loc@q##1##2{\textbf{Frage ##1} ##2}
154   \def\AMC@loc@corrected{Korrektur}
155   \def\AMC@loc@catalog{Katalog}
156   \def\AMC@loc@explain{\textit{\textbf{Erkl\"arung : }}}
157   \def\AMC@loc@none{Keine dieser Antworten ist korrekt.}
158   \def\AMC@loc@question{Frage}
159   \def\AMC@loc@questions{Fragen}
160   \def\AMC@loc@namesurname{Vor- und Nachname:}
161 }
```

4.4.6 Italian

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

```
162 \def\AMC@loc@IT{
163   \def\AMC@loc@draft{BOZZA}
164   \def\AMC@loc@message{Per l'esame, \e preferibile stampare i documenti
165     a partire da auto-multiple-choice.}
166   \def\AMC@loc@qf##1{\textbf{Domanda ##1:}}
167   \def\AMC@loc@q##1##2{\textbf{Domanda ##1} ##2}
168   \def\AMC@loc@corrected{Correzione}
169   \def\AMC@loc@catalog{Catalogo}
170   \def\AMC@loc@none{Nessuna risposta \e giusta.}
171   \def\AMC@loc@question{domanda}
172   \def\AMC@loc@questions{domande}
173   \def\AMC@loc@namesurname{Nome e cognome:}
174 }
```

4.4.7 Norwegian

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

```
175 \def\AMC@loc@NO{
176   \def\AMC@loc@draft{UTKAST}
177   \def\AMC@loc@message{Det anbefales {\aa} skrive ut dokumentet
178     for gjennomgang \direkte fra auto-multiple-choice.}
179   \def\AMC@loc@qf##1{\textbf{Oppgave ##1 :}}
180   \def\AMC@loc@q##1##2{\textbf{Oppgave ##1} ##2}
181   \def\AMC@loc@corrected{Rettet}
182   \def\AMC@loc@catalog{Katalog}
183   \def\AMC@loc@none{Ingen svar er riktige.}
184   \def\AMC@loc@question{oppgave}
185   \def\AMC@loc@questions{oppgave}
186   \def\AMC@loc@namesurname{Etternavn og fornavn:}
187 }
```

4.4.8 Portuguese

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

```

188 \def\AMC@loc@PT{
189   \def\AMC@loc@draft{RASCUNHO}
190   \def\AMC@loc@message{Para o seu exame, use preferencialmente documentos compilados do auto-multiple-choice}
191   \def\AMC@loc@qf##1{\textbf{Quest\~ao ##1:}}
192   \def\AMC@loc@q##1##2{\textbf{Quest\~ao ##1} ##2}
193   \def\AMC@loc@corrected{Corrigido}
194   \def\AMC@loc@catalog{Cat\'alogo}
195   \def\AMC@loc@explain{\textit{\textbf{Justifique: }}}
196   \def\AMC@loc@none{Nenhuma das respostas apresentadas est\'a correta.}
197   \def\AMC@loc@question{Quest\~ao}
198   \def\AMC@loc@questions{Quest\~oes}
199   \def\AMC@loc@namesurname{Nome e apelido:}
200 }

```

4.4.9 Spanish

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

```

201 \def\AMC@loc@ES{
202   \def\AMC@loc@draft{BORRADOR}
203   \def\AMC@loc@message{Para revisi'on, preferentemente imprimir documentos compilados
204     desde auto-multiple-choice.}
205   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
206   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
207   \def\AMC@loc@corrected{Correcci'on}
208   \def\AMC@loc@catalog{Cat'alogo}
209   \def\AMC@loc@none{Ninguna de estas preguntas son correctas.}
210   \def\AMC@loc@question{pregunta}
211   \def\AMC@loc@questions{preguntas}
212   \def\AMC@loc@namesurname{Nombre y apellidos:}
213 }

```

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).

```

214 \def\AMC@loc@JA{
215   \def\AMC@loc@draft{◇◆◆◆}
216   \def\AMC@loc@message{◇◆◆◆◆◆◆◆◆auto-multiple-choice◇◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆◆}
217   \def\AMC@loc@qf##1{\textbf{◇##1:}}
218   \def\AMC@loc@q##1##2{\textbf{◇##1} ##2}
219   \def\AMC@loc@corrected{◇◆◆◆}
220   \def\AMC@loc@catalog{◇◆◆◆◆◆}
221   \def\AMC@loc@explain{\textit{\textbf{◇◇: }}}
222   \def\AMC@loc@none{◇◆◆◆◆}
223   \def\AMC@loc@question{◇}
224   \def\AMC@loc@questions{◇}
225 }

```

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:

```
226 \AtBeginDocument{\@ifpackageloaded{cleveref}{%
227   \message{AMC/cleveref integration loaded^^J}%
228   \crefalias{AMCquestionaff}{question}%
229   \crefname{question}{\AMC@loc@question}{\AMC@loc@questions}%
230 }{}}%
```

4.6 Random

4.6.1 Random pseudo-generator

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

```
231 \ifx\AMC@SR\undefined\newcount\AMC@SR\fi
232 \providecommand\AMC@SRconst{2097152}
233 \providecommand\AMC@SRset[1]{\global\AMC@SR#1 \ignorespaces}
234 \providecommand\AMC@SRadvance{%
235   \begingroup%
236     \ifnum\AMC@SR<\AMC@SRconst\relax\AMC@SR@count\z@%else\AMC@SR@count\@ne\fi%
237     \ifodd\AMC@SR\advance\AMC@SR@count\@ne\fi%
238     \global\divide\AMC@SR\tw@%
239     \ifodd\AMC@SR@count\global\advance\AMC@SR\AMC@SRconst\relax\fi%
240   \endgroup}
241 \providecommand\AMC@SRbit{\AMC@SRadvance\ifodd\AMC@SR1\else0\fi}
242 \providecommand\AMC@SRtest[2]{\AMC@SRadvance%
243   \ifodd\AMC@SR#2\else#1\fi\ignorespaces}
244 \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>}`.

```
245 \AMC@SRset{1515}
246 \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.

```
247 \newcount\AMC@SR@count
248 \def\AMC@SR@time{\AMC@SRset{\time}}
249 \newcount\AMC@SRnum
250 \def\AMC@SRnextByte{\AMC@SRnum=\z@%
251   \AMC@SR@count=20%
252   \loop\multiply\AMC@SRnum\tw@%
253     \AMC@SRtest{\advance\AMC@SRnum\@ne}{}%
254   \ifnum\AMC@SR@count>\@ne\advance\AMC@SR@count\m@ne\repeat%
```

```

255 }
256 \newcommand\AMC@SRmax[1]{\AMC@SRnextByte%
257   \AMC@SR@count=\AMC@SRnum%
258   \divide\AMC@SR@count by #1\relax%
259   \multiply\AMC@SR@count by #1\relax%
260   \advance\AMC@SRnum by -\AMC@SR@count%
261 }

```

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).

```

262 \newcount\AMC@sti
263 \newcount\AMC@stil
264 \newtoks\AMCsw@p@
265 \newcommand\AMCsw@p[2]{%
266   \global\AMCsw@p@=#1%
267   \global#1=#2%
268   \global#2=\AMCsw@p@}
269 \newcommand{\AMC@shuffletoks}[3][\@ne]{%
270   \AMC@sti=#2\relax%
271   \AMC@stil=#2\relax%
272   \advance\AMC@stil\@ne%
273   \advance\AMC@stil -#1\relax%
274   \@whilenum\AMC@sti>#1\do{%
275     \AMC@SRmax{\AMC@stil}\advance\AMC@SRnum #1\relax%
276     \AMCsw@p{\csname #3\romannumeral\AMC@SRnum\endcsname}%
277       {\csname #3\romannumeral\AMC@sti\endcsname}%
278     \advance\AMC@sti\m@ne\relax%
279     \advance\AMC@stil\m@ne\relax%
280   }}

```

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).

```

281 \newcount\AMC@numerotation\AMC@numerotation=\z@%
282 \def\AMC@definitnumero#1#2{\AMCmessage{NUM=#1=#2}%
283   \expandafter\global\expandafter\def\csname AMC@numtab@#2\endcsname{#1}}
284 \def\AMC@prepare#1{\expandafter\ifx\csname AMC@numtab@#1\endcsname\relax%
285   \global\advance\AMC@numerotation\@ne%
286   \expandafter\AMC@definitnumero\expandafter{\the\AMC@numerotation}{#1}\fi}
287 \def\AMC@unnumero#1{\AMC@prepare{#1}\csname AMC@numtab@#1\endcsname}
288 \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.

```
289 \def\AMC@logchar#1#2{%
290   \protected@write\AMC@logfile{}{%
291     \string\answer%
292     {\the\AMCid@etud/\thepage:#2}%
293     {#1}}%
294 }
```

4.8.2 Position logging

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

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

```
295 \def\AMC@shapename@{\ifAMC@invisible none\else\AMC@shapename\fi}
296 \def\AMC@tracepos#1#2{%
297   \ifAMC@calibration\ifx\@empty#1\@empty\else%
298     \pdfsavepos\protected@write\AMC@XYFILE{}{%
299       \string\tracepos%
300       {\the\AMCid@etud/\thepage:#2}%
301       {\noexpand\number\pdflastxpos sp}%
302       {\noexpand\number\pdflastypos sp}%
303       {\AMC@shapename}}%
304   \fi\fi}
305 \def\AMC@traceposx#1#2{%
306   \ifAMC@calibration\ifx\@empty#1\@empty\else%
307     \pdfsavepos\protected@write\AMC@XYFILE{}{%
308       \string\tracepos%
309       {\the\AMCid@etud/\thepage:#2}%
310       {\noexpand\number\pdflastxpos sp}%
311       {0sp}%
312       {\AMC@shapename}}%
313   \fi\fi}
314 \def\AMC@traceposy#1#2{%
315   \ifAMC@calibration\ifx\@empty#1\@empty\else%
316     \pdfsavepos\protected@write\AMC@XYFILE{}{%
317       \string\tracepos%
318       {\the\AMCid@etud/\thepage:#2}%
319       {0sp}%
320       {\noexpand\number\pdflastypos sp}%
321       {\AMC@shapename}}%
322   \fi\fi}
323 \newcommand\AMC@tracebox[3]{%
324   \vbox{\AMC@traceposy{#1}{#2}%
325     \hbox{\AMC@traceposx{#1}{#2}#3\AMC@traceposx{#1}{#2}}%
326     \AMC@traceposy{#1}{#2}}%
327 \def\AMC@pagepos{%
```

```

328 \ifAMC@calibration\protected@write\AMC@XYFILE{}{%
329 \string\page%
330 {\the\AMCid@etud/\thepage/\the\AMCid@check}%
331 {\the\paperwidth}{\the\paperheight}%
332 {\the\pdfpagewidth}{\the\pdfpageheight}}\fi}

```

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

```

\AMCreTick 333 \newcommand{\AMCdontScan}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontscan{\the\AMCid@etud,\thepage,\the\AMCid@check}}}%
334 \newcommand{\AMCdontAnnotate}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontannotate{\the\AMCid@etud,\thepage,\the\AMCid@check}}}%
335 \newcommand{\AMCreTick}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\retick{\the\AMCid@etud,\thepage,\the\AMCid@check}}}%
336 %

```

\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.

```

337 \newcommand\AMC@tracechar[4]{%
338 \ifAMC@calibration\ifx\@empty#3\@empty\else%
339 \protected@write\AMC@XYFILE{}{%
340 \string\boxchar{\the\AMCid@etud/\thepage:#4}{#1}%
341 }%
342 \fi\fi%
343 }

```

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.

```

344 \newenvironment{amcxyfile}[1]{%
345 \protected@write\AMC@XYFILE{}{\string\xyopen{#1}}%
346 }{%
347 \protected@write\AMC@XYFILE{}{\string\xyclose{}}%
348 }

```

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

```

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

```

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

```

350 \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:

```

351 \newcommand{\namefielddots}{%
352 \noindent%
353 \ifAMC@pdfform%
354 \hspace*{\fill}%
355 \TextField[name={\the\AMCid@etud:namefield},width=.95\linewidth,bordercolor=0 0 0]{}%
356 \hspace*{\fill}%
357 \else%
358 \dotfill
359 \fi%
360 }

```

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}{1}` produces

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

8

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

8

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

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

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

```
366 \newbox\AMC@smashbox
367 \newdimen\AMC@smashboxheight
368 \newcommand{\AMC@smashcentered}[1]{%
```

```

369 \setbox\AMC@smashbox\hbox{#1}%
370 \AMC@smashboxheight=\ht\AMC@smashbox%
371 \advance\AMC@smashboxheight by \dp\AMC@smashbox%
372 \vfuzz=\AMC@smashboxheight\hfuzz=\wd\AMC@smashbox%
373 \hspace*{-.5\wd\AMC@smashbox}\hbox to .5\wd\AMC@smashbox{%
374 \vbox to 0pt{%
375 \vspace*{-.5\AMC@smashboxheight}\vbox to .5\AMC@smashboxheight{%
376 \box\AMC@smashbox}}}%
377 }}

```

`\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.

```

378 \newcommand\AMC@setcolors@[2]{%
379 \def\AMC@boxcolor@{\AMC@boxcolor}%
380 \ifx\@empty#1\@empty \def\AMC@boxcolor@{black}\fi%
381 \ifAMC@correc\def\AMC@boxcolor@{black}\fi%
382 \def\AMC@fillcolor@{\ifx #2\AMC@checkedbox%
383 \AMC@boxcolor@\else white\fi}%
384 \AMC@draw@crossfalse%
385 \ifKV@AMCdim@cross\ifx #2\AMC@checkedbox%
386 \AMC@draw@crosstrue\fi\fi%
387 }
388 \newcommand\AMC@answerBox@[4]{%
389 \ifAMC@catalog%
390 \AMC@logchar{#1}{#4}%
391 \fi%
392 \AMC@LR{\hspace{0pt}%
393 \lower\AMC@boxeddown\hbox{\csname AMC@shape@\AMC@shapename@\endcsname%
394 {AMCchoiceLabelFormat{#1}}{#2}{#3}{#4}}}%
395 }
396 \newcommand\AMC@shapeprepare@square{}
397 \newcommand\AMC@shape@square[4]{%
398 \fboxsep=\z@\fboxrule=\AMC@boxedrule%
399 \AMC@setcolors@{#3}{#2}%
400 \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
401 \fcolorbox{\AMC@boxcolor@}{\AMC@fillcolor@}%
402 {%
403 \boxput*(0,0){%
404 \ifAMC@draw@cross\AMC@crosschar\fi%
405 }{%
406 \vbox to \AMC@boxedheight{%
407 \AMC@tracepos{#3}{#4}%
408 \vfill%
409 \hbox to \AMC@boxedwidth{\hfill%
410 \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
411 \AMC@smashcentered{#2}%
412 \hfill}\vfill}}%
413 \AMC@tracepos{#3}{#4}}%
414 }

```

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

```

415 \newcommand\AMC@makeovalbox[3]{%
416 \AMC@setcolors@{#1}{#2}%

```

```

417 \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
418 \AMC@save@box{#3}{%
419 \begin{tikzpicture}%
420 \useasboundingbox (-0.5\AMC@boxedwidth-0.5\AMC@boxedrule,0.5\AMC@boxedheight+0.5\AMC@boxedrule)
421 rectangle (0.5\AMC@boxedwidth+0.5\AMC@boxedrule,-0.5\AMC@boxedheight-0.5\AMC@boxedrule);
422 \draw[\AMC@boxcolor@,fill=\AMC@fillcolor@,line width=\AMC@boxedrule,rounded corners=\AMC@oval@radi
423 (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight)
424 rectangle (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
425 \ifAMC@draw@cross
426 \draw[\AMC@boxcolor@,line width=\AMC@crossrule]
427 (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight)
428 (0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (-0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
429 \fi
430 \end{tikzpicture}}%
431 }
432 \newcommand\AMC@shapeprepare@oval{%
433 \ifx\AMC@ovalbox@R\@undefined\else%
434 \AMC@makeovalbox{1}{\AMC@ovalbox@R}%
435 \AMC@makeovalbox{1}{\AMC@checkbox}{\AMC@ovalbox@RF}%
436 \AMC@makeovalbox{}{\AMC@ovalbox@}%
437 \AMC@makeovalbox{}{\AMC@checkbox}{\AMC@ovalbox@F}%
438 \fi%
439 }
440 \newcommand\AMC@shape@oval[4]{%
441 \AMC@setcolors@{#3}{#2}%
442 \AMC@tracebox{#3}{#4}{\boxput*(0,0){%
443 \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
444 \AMC@smashcentered{#2}%
445 }{%
446 \ifx\@empty#3\@empty%
447 \ifx #2\AMC@checkbox%
448 \AMC@use@box{\AMC@ovalbox@F}%
449 \else%
450 \AMC@use@box{\AMC@ovalbox@}%
451 \fi%
452 \else%
453 \ifx #2\AMC@checkbox%
454 \AMC@use@box{\AMC@ovalbox@RF}%
455 \else%
456 \AMC@use@box{\AMC@ovalbox@R}%
457 \fi%
458 \fi%
459 }}%
460 }
461 \newcommand\AMC@shapeprepare@form{}
462 \newcommand\AMC@shape@form@base[5]{%
463 \ifx #2\AMC@checkbox%
464 \def\AMC@shape@form@ticked{true}%
465 \else%
466 \def\AMC@shape@form@ticked{false}%
467 \fi%
468 \AMC@tracebox{#3}{#4}{%
469 \CheckBox[checked=\AMC@shape@form@ticked,%

```

```

470         checkboxsymbol=\ding{110},name={#5},%
471         bordercolor=0 0 0,
472         width=\AMC@boxedwidth,height=\AMC@boxedheight]{#3}{#4}%
473     }%
474 }
475 \newcommand\AMC@shape@form[4]{%
476     \AMC@shape@form@base{#1}{#2}{#3}{#4}{\the\AMCid@etud:#4}%
477 }
478 \newcommand\AMC@shapeprepare@none{}
479 \newcommand\AMC@shape@none[4]{#1}

```

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

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

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

```

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

```

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

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

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

```

491 \newlength\AMC@boxedrule
492 \newlength\AMC@crossrule
493 \newlength\AMC@boxeddown
494 \newlength\AMC@boxedwidth
495 \newlength\AMC@boxedheight
496 \newlength\AMC@oval@radius
497 \newlength\AMC@outside@sep
498 \define@choicekey{AMCdim}{shape}{square,oval,form,none}{\def\AMC@shapename{#1}}
499 \define@key{AMCdim}{size}{\AMC@boxedwidth=#1\AMC@boxedheight=#1}
500 \define@key{AMCdim}{height}{\AMC@boxedheight=#1}
501 \define@key{AMCdim}{width}{\AMC@boxedwidth=#1}

```

```

502 \define@key{AMCdim}{rule}{\AMC@boxedrule=#1}
503 \define@key{AMCdim}{outsidesep}{\AMC@outside@sep=#1}
504 \define@key{AMCdim}{down}{\AMC@boxeddown=#1}
505 \define@key{AMCdim}{color}{\def\AMC@boxcolor{#1}}
506 \define@boolkey{AMCdim}{cross}[false]{}
507 \define@key{AMCdim}{crosschar}{\textbf{\textsf{X}}}{\def\AMC@crosschar{#1}}
508 \define@key{AMCdim}{crossrule}[1.5pt]{\AMC@crossrule=#1}
509 \def\AMC@shapeprepare{\csname AMC@shapeprepare@\AMC@shapename@ \endcsname}
510 \def\AMCboxStyle#1{%
511   \setkeys{AMCdim}{#1}%
512   \ifnum\AMC@boxedwidth<\AMC@boxedheight%
513     \AMC@oval@radius=\AMC@boxedwidth\divide\AMC@oval@radius\tw@%
514   \else%
515     \AMC@oval@radius=\AMC@boxedheight\divide\AMC@oval@radius\tw@%
516   \fi%
517   \AMC@shapeprepare%
518 }
519 \AMCboxStyle{shape=square,size=2.5ex,down=.4ex,rule=.5pt,outsidesep=.1em,color=black,cross,crosschar,crossrule}
520 \newcommand\AMCboxColor[1]{\AMCboxStyle{color=#1}}
521 \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`).

```

522 \def\AMCoutsideLabelFormat#1{#1}
523 \newcommand\AMCboxOutsideLetter[2]{#1\nobreak\hspace{.1em}\AMCoutsideLabelFormat{#2}}
524 \newif\ifAMC@printformoutside@%
525 \newcommand\ifAMC@printformoutside{%
526   \AMC@printformoutside@false%
527   \ifAMC@ensemble\ifAMC@outside@box%
528     \ifAMCformulaire@dedans\AMC@printformoutside@true\fi%
529     \ifAMC@zoneformulaire\AMC@printformoutside@true\fi%
530   \fi\fi%
531   \ifAMC@printformoutside@%
532 }
533 \newcommand\AMC@formBox@[4]{%
534   \ifAMC@printformoutside% letter to be written outside the box
535     \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
536   \else%
537     \AMC@answerBox@{#1}{#2}{#3}{#4}%
538   \fi%
539   \AMC@tracechar{#1}{#2}{#3}{#4}%
540 }
541 \newif\ifAMC@printkeyoutside@%
542 \newcommand\ifAMC@printkeyoutside{%

```

```

543 \AMC@printkeyoutside@false%
544 \ifAMC@ensemble%
545 \ifAMC@outside@box\AMC@printkeyoutside@true\fi%
546 \else%
547 \ifAMC@inside@box\else\AMC@printkeyoutside@true\fi%
548 \fi%
549 \ifAMC@printkeyoutside@%
550 }
551 \newcommand\AMC@keyBox@[4]{%
552 \ifAMC@printkeyoutside%
553 \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
554 \else%
555 \AMC@answerBox@{#1}{#2}{#3}{#4}%
556 \fi%
557 \AMC@tracechar{#1}{#2}{#3}{#4}%
558 }
559 \newcommand\AMC@formBox[4]{%
560 \AMC@formBox@{\ifx\@empty#1\@empty%
561 \AMCchoiceLabel{AMC@ncase}%
562 \else #1\fi}{#2}{#3}{#4}%
563 }
564 \newcommand{\AMC@box}[2]{%
565 \ifAMC@ensemble%
566 \ifAMC@zoneformulaire% for codes inside form sheet
567 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
568 \else%
569 \ifAMC@formulaire@dedans% for answer boxes inside form sheet
570 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
571 \else% outside form sheet: not to be read during data capture
572 \AMC@formBox{#1}{#2}{1}{casequestion:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
573 \fi\fi%
574 \else% no separate sheet for answers: always read
575 \ifAMC@inside@box%
576 \AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
577 \else%
578 \AMC@formBox@{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
579 \fi%
580 \fi%
581 }

```

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.

```

582 \newif\ifAMCsz@logged\AMCsz@loggedfalse
583 \newcommand{\AMCscoreZone}[1]{%
584 \ifAMC@ensemble%
585 \ifAMC@formulaire@dedans%
586 \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
587 \else%
588 \AMC@tracebox{1}{scorequestion::\the\AMCid@quest,-1}{#1}%
589 \fi%
590 \else%

```



```

591 \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
592 \fi%
593 \ifAMCsz@logged\else%
594 \AMCmessage{VAR:scorezones=1}%
595 \global\AMCsz@loggedtrue%
596 \fi%
597 }

```

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`.

```

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

```

`\AMC@binaryCode` The command `\AMC@binaryCode{<options>}{<n>}` prints boxes to represent the number $\langle n \rangle$ 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:



```

604 \newtoks\AMCbin@sequence
605 \newcount\AMCbin@number
606 \newcount\AMCbin@digit
607 \newcount\AMCbin@id
608 \newcount\AMCbin@did
609 \newcount\AMCbin@ndigits
610 \newdimen\AMCbin@hsep
611 \define@key{AMCbin}{ndigits}{\AMCbin@ndigits=#1}
612 \define@key{AMCbin}{id}{\AMCbin@id=#1}
613 \define@key{AMCbin}{hsep}{\AMCbin@hsep=#1}
614 \define@key{AMCbin}{style}[]{\def\AMCbin@style{#1}}

```

```

615 \def\AMCbin@one{%
616   \ifnum\AMCbin@did>\z@%
617     \hspace{\AMCbin@hsep}%
618     \fi%
619     \advance\AMCbin@did\@ne%
620   \ifnum\AMCbin@id>0%
621     \AMC@answerBox@{\AMC@checkbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
622     \else%
623     \AMC@answerBox@{\AMC@checkbox}{1}{}%
624     \fi}
625 \def\AMCbin@zero{%
626   \ifnum\AMCbin@did>\z@%
627     \hspace{\AMCbin@hsep}%
628     \fi%
629     \advance\AMCbin@did\@ne%
630   \ifnum\AMCbin@id>0%
631     \AMC@answerBox@{\AMC@checkbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
632     \else%
633     \AMC@answerBox@{\AMC@checkbox}{1}{}%
634     \fi}
635 \newcommand{\AMC@binaryCode}[2]{%
636   \setkeys{AMCbin}{ndigits=1,hsep=0pt,style}\setkeys{AMCbin}{#1}%
637   \AMCbin@did=\z@%
638   {\AMCboxDimensions{shape=square,size=.32cm,down=0pt,rule=.2pt,cross=false}\expandafter\AMCboxDimensions}%
639   \AMCbin@digit=\z@%
640   \loop%
641   \ifnum\AMCbin@number>\z@%
642     \advance\AMCbin@digit\@ne%
643     \ifodd\AMCbin@number\AMCbin@sequence=\expandafter{\expandafter\AMCbin@one\the\AMCbin@sequence}%
644     \else\AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\fi%
645     \divide\AMCbin@number\two%
646     \repeat%
647   \loop\relax%
648   \ifnum\AMCbin@digit<\AMCbin@ndigits\advance\AMCbin@digit\@ne%
649   \AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\repeat%
650   \the\AMCbin@sequence%
651   \ifnum\AMCbin@digit>\AMCbin@ndigits\PackageError{automultiplechoice}{Too low AMC@NCB value (got \the\AMCbin@digit)}{}
652 }

```

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

```

653 \def\AMCbin@begin#1{\setkeys{AMCbin}{id=#1}}
654 \newcommand{\AMC@binaryBoxes}[2][1]{%
655   \AMC@binaryCode{ndigits=#1}{#2}%
656 }

```

4.9 Checking Environment

`\AMCcurrentenv` Sets the current environment as document.

```
657 \def\AMCcurrentenv{document}
```

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

```
658 \def\AMCif@env#1{
```

```

659 \def\AMC@tempenv{#1}%
660 \ifx\AMC@tempenv\AMCcurrentenv
661 \expandafter\@firstoftwo
662 \else
663 \expandafter\@secondoftwo
664 \fi
665 }

```

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.

```

666 \newcount\AMCtok@k
667 \newcount\AMCtok@max
668 \newcount\AMCtok@size
669 \newcommand{\nouveaugroupe}[2]{%
670 \expandafter\ifx\csname #1@k\endcsname\relax%
671 \expandafter\newcount\csname #1@k\endcsname%
672 \expandafter\newcount\csname AMC#1@j\endcsname%
673 \csname #1@k\endcsname=\z@\relax%
674 \csname AMC#1@j\endcsname=\z@\relax%
675 \setgroupmode{#1}{\AMCdefaultgroupmode}%
676 \fi%
677 }
678 \newcommand\AMC@prepare@element[1]{%
679 \nouveaugroupe{#1}{}%
680 \global\advance\csname #1@k\endcsname\@ne\relax%
681 \AMCtok@k=\csname #1@k\endcsname%
682 \expandafter\ifx\csname #1@\romannumeral\AMCtok@k\endcsname\relax%
683 \expandafter\newtoks\csname #1@\romannumeral\AMCtok@k\endcsname\fi%
684 }
685 \newcommand{\element}[2]{%
686 \AMC@prepare@element{#1}%
687 \global\csname #1@\romannumeral\AMCtok@k\endcsname={#2}%
688 }

```

\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.

```

689 \def\AMCdefault@groupmode{fixed}
690 \newcommand{\setdefaultgroupmode}[1]{\def\AMCdefault@groupmode{#1}}
691 \newcommand{\setgroupmode}[2]{%
692   \expandafter\ifx\csname AMCgrouppre@#2\endcsname\relax%
693     \PackageError{automultiplechoice}{Unknown group mode for #1 : #2}%
694     {You asked to set group '#1' mode to '#2',
695      but '#2' is not a valid group mode}%
696   \else%
697     \expandafter\global\expandafter\def\csname AMC#1@mode\endcsname{#2}%
698   \fi%
699 }
```

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

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

```

700 \newcommand{\AMCgrouppre@fixed}[3]{%
701   \ifnum#3<\z@%
702     \csname AMC#1@j\endcsname=\z@%
703   \else%
704     \csname AMC#1@j\endcsname=#3%
705   \fi%
706 }
```

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

```

707 \newcommand{\AMCgrouppre@withreplacement}[3]{%
708   \ifnum#3<\z@%
709     \csname AMC#1@j\endcsname=\z@%
710   \else%
711     \csname AMC#1@j\endcsname=#3%
712   \fi%
713   \shufflegroup{#1}%
714 }
```

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.

```

715 \newcount\AMC@imax
716 \newcommand{\AMCgrouppre@withoutreplacement}[3]{%
717   \ifnum#3<\z@%
718   \else%
719     \csname AMC#1@j\endcsname=#3%
720   \fi%
721   \ifnum\AMCtok@ik=\AMCloop@k%
722     \AMCtok@ik=\z@%
723   \fi%
```

```

724 \ifnum\AMCtok@ik=\z0%
725   \shufflegroup{#1}%
726 \else%
727   \AMC@imax=\AMCloop@k%
728   \advance\AMC@imax -#2\relax%
729   \ifnum\AMCtok@ik>\AMC@imax%
730     \shufflegroupslice{#1}{\@ne}{\AMCtok@ik}%
731     \ifnum\AMCtok@ik<\AMCloop@k%
732       \advance\AMCtok@ik\@ne%
733       \shufflegroupslice{#1}{\AMCtok@ik}{\AMCloop@k}%
734     \fi%
735   \fi%
736 \fi%
737 }

```

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

```

738 \newcommand{\AMCgrouppre@cyclic}[3]{%
739   \ifnum#3<\z0%
740   \else%
741     \csname AMC#1@j\endcsname=#3%
742   \fi%
743 }

```

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

```

744 \newcommand{\AMCgroup@pre}[4]{%
745   \csname AMCgrouppre@#1\endcsname{#2}{#3}{#4}%
746 }

```

`\shufflegroup` Command `\shufflegroup{<group-name>}` shuffles the elements of group `<group-name>`, and

`\insertgroup` `\shufflegroupslice{<group-name>}{<a>}{}` shuffles elements `<a>` to `` from group `<group-name>`.

`\insertgroupfrom` It can be called at each student sheet in order to get different student sheets and avoid cheating.

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

```

747 \newcommand{\shufflegroup}[1]{%
748   \ifAMC@shuffleG{\AMC@shuffletoks{\number\csname #1@k\endcsname}{#1@}}\fi%
749 }
750 \newcommand{\shufflegroupslice}[3]{%
751   \ifAMC@shuffleG{\AMC@shuffletoks[#2]{#3}{#1@}}\fi%
752 }
753 \newcount\AMCtok@ik
754 \newcount\AMCloop@k
755 \newcommand{\AMCgrouploop@prep}[3]{%
756   \AMCtok@size=#1\relax%
757   \ifAMC@fullGroups\AMCtok@size=\m@ne\fi%
758   \ifnum\AMCtok@size<\z0%
759     \AMCtok@size=\csname #2@k\endcsname%
760   \fi%
761   \AMCtok@ik=\csname AMC#2@j\endcsname%
762   \AMCloop@k=\csname #2@k\endcsname%
763   \expandafter\ifx\csname AMC#2@mode\endcsname\relax%
764     \PackageError{automultiplechoice}{No group mode for #2}%

```

```

765      {No mode has been defined for group '#2'. This should not occur...}%
766    \fi%
767    \AMCgroup@pre{\csname AMC#2@mode\endcsname}{#2}{\the\AMCtok@size}{#3}%
768  }
769  \newcommand{\AMCgrouploop@next}[1]{%
770    \global\advance\csname AMC#1@j\endcsname\@ne\relax%
771    \expandafter\ifnum\csname AMC#1@j\endcsname>\AMCloop@k\relax%
772      \global\csname AMC#1@j\endcsname=\@ne%
773    \fi%
774    \AMCtok@ik=\csname AMC#1@j\endcsname%
775    \advance\AMCtok@size\m@ne%
776  }
777  \newcommand{\insertgroupfrom}[3][-1]{%
778    \ifnum#1=0%
779      \else%
780        \AMCgrouploop@prep{#1}{#2}{#3}%
781        {\loop%
782          \AMCgrouploop@next{#2}%
783          {\the\csname #2@\romannumeral\AMCtok@ik\endcsname}%
784          \ifnum\AMCtok@size>\z@\repeat}%
785        \fi%
786      }
787    \newcommand{\insertgroup}[2][-1]{%
788      \insertgroupfrom{#1}{#2}{-1}%
789    }

```

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

`\cleargroup{<group>}` clears the group `<group>`, erasing all of its elements.

`\copygroup[<n>]{<from>}{<to>}` copies `<n>` elements from group `<from>` to group `<to>`. If optional parameter `<n>` is not given, all the questions from group `<from>` are copied. `\copygroupfrom[<n>]{<from>}{<to>}{<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.

```

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

```

```

807 \copygroupfrom[#1]{#2}{#3}{-1}%
808 }

```

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`.

```

809 \newcount\AMCrep@count
810 \newcount\AMCrep@nn
811 \newcount\AMCrep@nnmax
812 \AMCload@counter=199
813 \@whilenum\AMCload@counter>0\do{%
814   \expandafter\newtoks\csname reponse@\romannumeral\AMCload@counter\endcsname%
815   \advance\AMCload@counter\m@ne%
816 }

```

`\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>`.

```

817 \newcommand\AMCload@reponse[2]{%
818   \global\advance\AMCload@counter\@ne\relax%
819   \global\csname reponse@\romannumeral\AMCload@counter\endcsname%
820   =\expandafter{\expandafter\AMCrep@count\expandafter=#2 #1}%
821 }
822 \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.

```

823 \def\shuffle@it{\AMC@shuffletoks{\number\AMCload@counter}{reponse@}}
824 \newcount\AMCnum@questions
825 \newcommand\AMCdump@reponses{%
826   \global\AMCnum@questions=\AMCload@counter%
827   \@whilenum\AMCload@counter>0\do{%
828     \the\csname reponse@\romannumeral\AMCload@counter\endcsname%
829     \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 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`.

```

830 \newcommand\AMCrep@init[1]{%
831   \ifAMC@ordre\AMCrep@o\else%
832     \csname AMCrep@#1\endcsname\fi\AMCload@counter=\z@}
833 \newcommand\AMCrep@of{%
834   \def\AMCload@reponse{\AMCrien@deux}\def\AMCrep@fini{}}
835 \newcommand\AMCrep@r{%
836   \def\AMCload@reponse{\AMCload@reponse}%
837   \def\AMCrep@fini{\shuffle@it\AMCdump@reponses}}
838 \newcount\AMCrep@count
839 \newcommand\lastchoices{%
840   \AMCrep@count=\AMCrep@count%
841   \AMCrep@fini\AMCrep@init{o}%
842   \AMCrep@count=\AMCrep@count}
843 \newcommand\@aucune{\emph{\AMC@loc@none}}
844 \newcommand\AMC@fin@rep{%
845   \ifAMCcomplete@multi\ifAMCtype@multi%
846     \lastchoices\AMCrep@count=-1%
847     \ifAMC@bonne\wrongchoice{\@aucune}\else%
848       \ifAMC@postcorrect\wrongchoice{\@aucune}\else\correctchoice{\@aucune}\fi%
849     \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.

```

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

```



```

863 \csname\AMC@sza@callin\endcsname%
864 \fi%
865 }
866 \def\AMCformAnswer#1{\hspace{\AMCformHSpace} #1}
867 \def\AMCformAnswerA#1{\addtocounter{AMC@ncase}{1}\AMCformAnswer{#1}}

```

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

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

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

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

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

```

868 \ExplSyntaxOn
869
870 \prg_set_conditional:Nnn \amc_if_separate_question: { p , T } {
871   \ifAMC@ensemble
872     \ifAMC@zoneformulaire
873       \prg_return_false:
874     \else
875       \prg_return_true:
876     \fi
877   \else
878     \prg_return_false:
879   \fi
880 }
881 \cs_new_eq:NN \AMC@if@separate@question \amc_if_separate_question:T
882
883 \int_new:N \amc_memory_elts_count
884
885 \cs_new:Nn \amc_clear_memory: { \int_gzero:N \amc_memory_elts_count }
886 \cs_new_eq:NN \AMC@mem@clear \amc_clear_memory:
887
888 \cs_new:Npn \amc_memory_elt_i:n #1 {
889   amc_memory_elts_ \int_to_alph:n { #1 }
890 }
891 \cs_new:Nn \amc_memory_current_elt: {
892   \amc_memory_elt_i:n \amc_memory_elts_count
893 }
894 \cs_new:Npn \amc_memory_vars_i:n #1 {
895   amc_memory_vars_ \int_to_alph:n { #1 }
896 }
897 \cs_new:Nn \amc_memory_current_vars: {
898   \amc_memory_vars_i:n \amc_memory_elts_count
899 }
900
901 \cs_new:Nn \amc_add_memory_elt: {
902   \int_gincr:N \amc_memory_elts_count
903   \tl_gclear_new:c { \amc_memory_current_elt: }
904   \tl_gclear_new:c { \amc_memory_current_vars: }
905 }

```

```

906 \cs_new_eq:NN \AMC@mem@next \amc_add_memory_elt:
907
908 \cs_new:Npn \amc_add_to_memory:n #1 {
909   \tl_gput_right:cn { \amc_memory_current_elt: } { #1 }
910 }
911 \cs_new_eq:NN \AMC@mem@add \amc_add_to_memory:n
912
913 \cs_new:Npn \amc_add_to_vars:n #1 {
914   \tl_gput_right:cn { \amc_memory_current_vars: } { #1 }
915 }
916 \cs_new_eq:NN \AMC@mem@addvar \amc_add_to_vars:n
917
918 \cs_new:Npn \amc_add_qidaffname:nnn #1#2#3 {
919   \amc_add_to_vars:n {\AMCid@quest=#1\setcounter{AMCquestionaff}{#2}%
920     \global\def\AMCid@name{#3}}
921 }
922 \cs_generate_variant:Nn \amc_add_qidaffname:nnn { xxx }
923 \cs_new_eq:NN \AMC@mem@qidaffname \amc_add_qidaffname:xxx
924
925 \cs_new:Npn \amc_mem_elt_cat:n #1 {
926   \amc_add_to_vars:n { \def\AMCmem@elt@cat{ #1 } }
927 }
928 \cs_generate_variant:Nn \amc_mem_elt_cat:n { x }
929 \cs_new_eq:NN \AMC@mem@category \amc_mem_elt_cat:x
930
931 \cs_new:Npn \amc_add_aid:n #1 {
932   \amc_add_to_memory:n {\AMCrep@count=#1}
933 }
934 \cs_generate_variant:Nn \amc_add_aid:n { x }
935 \cs_new_eq:NN \AMC@mem@aid \amc_add_aid:x
936
937 \cs_new:Npn \amc_if_category_is_p:n #1 {
938   \str_if_eq_p:on { \AMCmem@elt@cat } { #1 }
939 }
940 \cs_new:Npn \amc_use_memory:n #1 {
941   \int_step_inline:nnnn { 1 } { 1 } \amc_memory_elts_count {
942     \def\AMCmem@elt@cat{ plain }
943     \tl_use:c { \amc_memory_vars_i:n { ##1 } }
944     \bool_if:nTF { #1 } {
945       \tl_use:c { \amc_memory_elt_i:n { ##1 } }
946     } { }
947   }
948 }
949 \cs_new:Nn \amc_use_memory: { \amc_use_memory:n { \c_true_bool } }
950 \cs_new_eq:NN \AMC@mem@show \amc_use_memory:
951 \cs_new_eq:NN \AMC@mem@show@filter \amc_use_memory:n
952 \cs_new_eq:NN \AMC@ifcategory \amc_if_category_is_p:n
953
954 \ExplSyntaxOff
955 \newcommand\AMC@mem@add@ifneeded[1]{%
956   \AMC@if@separate@question{%
957     \AMC@mem@add{#1}%
958   }%

```

```

959 }
960 \newcommand\AMC@mem@addsingle@ifneeded[2]{%
961   \AMC@if@separate@question{%
962     \AMC@mem@next%
963     \AMC@mem@category{#2}%
964     \AMC@mem@add{#1}%
965   }%
966 }
967 \newcommand\AMC@mem@answer[1]{%
968   \addtocounter{AMC@ncase}{1}%
969   \AMC@if@separate@question{%
970     \AMC@mem@aid{the\AMCrep@count}%
971     \AMC@mem@add{\AMCformAnswerA{#1}}%
972   }%
973 }
974 \newcommand\AMC@mem@openQuestion{%
975   \AMC@if@separate@question{%
976     \AMC@mem@next%
977     \AMC@mem@qidaffname{the\AMCid@quest}{\arabic{AMCquestionaff}}{\AMCid@name}%
978     \AMC@mem@add{\AMCformQuestionA}%
979   }%
980 }
981 \def\AMCformBegin{%
982   \AMC@zoneformulairetrue\setcounter{section}{0}%
983   \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageFull}\fi\fi%
984 }
985 \newcommand\AMCform{%
986   \ifAMC@ensemble\AMCformulaire@dedanstrue%
987     \AMC@mem@show%
988   \fi}
989 \newcommand\AMCformFilter[1]{%
990   \ifAMC@ensemble\AMCformulaire@dedanstrue%
991     \AMC@mem@show@filter{#1}%
992   \fi}
993 \newif\ifAMC@keepmemory
994 \newcommand\AMCforms{%
995   \ifAMC@ensemble\AMCformulaire@dedanstrue%
996     \AMCmessage{BR=0}\AMC@mem@show%
997     \global\AMC@keepmemorytrue%
998   \fi}

```

`\AMCtableForm` `\AMCtableForm[$\langle options \rangle$]` is a variant of `\AMCform` that displays the boxes as a multi-column table.

The argument $\langle options \rangle$ is a key-value list, where:

- **nanswers**= $\langle n \rangle$ gives the number of answers that are labeled in the table (defaults to the maximal number of answers of the subject).
- **ncols**= $\langle n \rangle$ gives the number of columns to use (defaults to the largest possible number of columns).
- **idtext**= $\langle text \rangle$ gives a text to be inserted before each question number.
- **idtitle**= $\langle text \rangle$ gives a text to be used as a column title for question numbers.

- `columnsep=<dim>` and `columnseprule=<dim>` are passed to the `multicols` environment.
- `headers=<bool>` tells if the headers with boxes letters are to be added at the top of the columns (defaults to true).
- `inside=<bool>` tells if letters are to be written inside the boxes (defaults to false).

```

999 \newcounter{AMC@specsi}
1000 \def\AMC@formspecs#1{%
1001   \noindent\hbox to \AMCformTH{\hspace*{\fill}\textbf{\AMC@tf@idtitle}\hspace{\AMCformHSpace}}%
1002   \setcounter{AMC@specsi}{0}%
1003   \loop%
1004     \addtocounter{AMC@specsi}{1}%
1005     \AMCformAnswer{\bf\AMCchoiceLabel{AMC@specsi}}%
1006     \ifnum\value{AMC@specsi}<#1\repeat%
1007 }
1008 \newcount\AMC@coli
1009 \def\AMC@formcolspecs#1#2{%
1010   \columnseprule=Opt%
1011   \begin{multicols}{#1}
1012     \AMC@coli=\z@%
1013     \@whilenum\AMC@coli<#1\do{
1014       \advance\AMC@coli\@ne%
1015       \AMC@formspecs{#2}\par
1016     }
1017   \end{multicols}
1018 }}
1019 \newcount\AMC@tf@ncols
1020 \newdimen\AMCformTH
1021 \newdimen\AMC@tf@colw
1022 \newdimen\AMC@tf@faw
1023 \def\AMC@tf@idtext{}
1024 \def\AMC@tf@idtitle{}
1025 \newbox\AMC@tf@box
1026 \define@key{AMCtf}{nanswers}[0]{\def\AMC@tf@nanswers{#1}}
1027 \define@key{AMCtf}{ncols}[0]{\AMC@tf@ncols=#1}
1028 \define@key{AMCtf}{idtext}{\def\AMC@tf@idtext{#1}}
1029 \define@key{AMCtf}{idtitle}{\def\AMC@tf@idtitle{#1}}
1030 \define@key{AMCtf}{idwidth}[Opt]{\AMCformTH=#1}
1031 \define@key{AMCtf}{columnsep}[0.5em]{\columnsep=#1}
1032 \define@key{AMCtf}{columnseprule}[0.5pt]{\columnseprule=#1}
1033 \define@boolkey{AMCtf}{headers}[true]{}
1034 \define@boolkey{AMCtf}{inside}[false]{}
1035 \setkeys{AMCtf}{nanswers,ncols,idwidth,headers,inside}
1036 \newcommand\AMCtableForm[1][{}]{%
1037   \setkeys{AMCtf}{columnsep,columnseprule}
1038   \setkeys{AMCtf}{#1}%
1039   \ifnum\AMC@tf@nanswers=0%
1040     \def\AMC@tf@nanswers{\the\AMCrep@nnmax}%
1041     \fi%
1042   \newdimen\AMC@tf@colW\AMC@tf@colW=\AMCformHSpace%
1043   \advance\AMC@tf@colW by \AMC@boxedwidth%
1044   \def\AMCformBeforeQuestion{\vspace{\AMCformVSpace}\par}%
1045   \def\AMCformQuestion##1{\noindent\hbox to \AMCformTH{\hspace*{\fill}\textbf{\AMC@tf@idtext{##1}}\hspa

```

```

1046 \def\AMCformAnswer##1{\hbox to \AMC@tf@colW{\hspace*{\fill}##1\hspace*{\fill}}}%
1047 \ifnum\AMCformTH=0%
1048   \setbox\AMC@tfbox=\hbox{\textbf{\AMC@tf@idtext{}}999}}%
1049   \AMCformTH=\wd\AMC@tfbox%
1050 \fi%
1051 \ifnum\AMC@tf@ncols=0%
1052   \AMC@tfcolw=\AMC@tf@nanswers\AMC@tf@colW%
1053   \advance\AMC@tfcolw by \AMCformTH%
1054   \advance\AMC@tfcolw by \columnsep%
1055   \AMC@tfaw=\linewidth%
1056   \loop%
1057   \ifnum\AMC@tfaw>\AMC@tfcolw%
1058     \advance\AMC@tfaw by -\AMC@tfcolw%
1059     \advance\AMC@tf@ncols\@ne%
1060   \repeat%
1061 \fi%
1062 \ifKV@AMCtf@headers%
1063   \AMC@formcolspecs{\the\AMC@tf@ncols}{\AMC@tf@nanswers}%
1064   \vspace{\AMCformVSpace}\vspace{-2\multicolsep}
1065 \fi%
1066 {\raggedcolumns
1067   \begin{multicols}{\the\AMC@tf@ncols}
1068     \ifKV@AMCtf@inside\else\def\AMCchoiceLabel##1{\fi%
1069       \AMCform
1070     \end{multicols}}
1071 }

```

`\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.

```

1072 \newcommand{\AMCsectionNumbered}[1]{%
1073   \section{#1}\AMC@mem@addsingle@ifneeded{\section{#1}}{section}}
1074 \newcommand{\AMCsubsectionNumbered}[1]{%
1075   \subsection{#1}\AMC@mem@addsingle@ifneeded{\subsection{#1}}{subsection}}
1076 \newcommand{\AMCsectionStar}[1]{%
1077   \section*{#1}\AMC@mem@addsingle@ifneeded{\section*{#1}}{section}}
1078 \newcommand{\AMCsubsectionStar}[1]{%
1079   \subsection*{#1}\AMC@mem@addsingle@ifneeded{\subsection*{#1}}{subsection}}
1080 \def\AMCsection{\@ifstar\AMCsectionStar\AMCsectionNumbered}
1081 \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 `\MCBoxedAnswers` 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.) 1082 \def\MCBoxedAnswers{\AMC@rbloctrue}
\MCBoxedAnswers 1083 \newcommand{\start@Answers}{%
1084   \global\AMCrep@count=\z@%
1085   \global\AMCrep@nn=\z@%
1086 }
1087 \newenvironment{choices}[1][r]{%
1088   \def\une@rep{\AMCrep@itemize}%

```

```

1089 \ifAMC@rbloc\def\une@rep{\AMCrep@bloc}%
1090 \else\begin{itemize}\setlength{\itemsep}{\AMCinterIrep}\fi%
1091 \AMCrep@init{#1}}%
1092 {\AMC@fin@rep\ifAMC@rbloc\else\end{itemize}\fi}
1093 \newenvironment{choiceshoriz}[1][r]{%
1094 \def\une@rep{\AMCrep@ligne}\AMCrep@init{#1}%
1095 \par\begin{center}}%
1096 {\AMC@fin@rep\end{center}}
1097 \newenvironment{choicescustom}[1][r]{%
1098 \def\une@rep{\AMCrep@perso}\AMCrep@init{#1}%
1099 \AMCbeginAnswer\ignorespaces}%
1100 {\AMC@fin@rep\AMCendAnswer}
1101 \newenvironment{tikz-single}[1][r]{
1102 \def\une@rep{\AMCrep@tikz}\AMCrep@init{#1}%
1103 \begin{tikzpicture}}{\AMC@fin@rep\end{tikzpicture}}
1104 \newenvironment{tikz-multi}[1][r]{
1105 \def\une@rep{\AMCrep@tikzmat}\AMCrep@init{#1}%
1106 \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
`\AMCrep@tikz` format the answer with a box given in `⟨box⟩` and text `⟨text⟩`. `\AMCrep@bloc` is also defined and
`\AMCrep@tikzmat` used in standard formatting when the user wants to put answers inside a L^AT_EX box.

```

\AMCrep@itemize 1107 \newcommand\AMCrep@bloc[2]{\AMC@mem@answer{#1}%
\AMCrep@ligne 1108 \par%
\AMCrep@perso 1109 \ifAMC@textPos\vbox\bgroup\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\hbox{
1110 \noindent\begin{minipage}{\linewidth}%
1111 \begin{itemize}\item[1] #2\end{itemize}\end{minipage}%
1112 \ifAMC@textPos\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\egroup\AMC@tra
1113 \vspace{\AMCinterBrep}}
1114 \newcommand\AMCrep@tikz[5]{\AMC@mem@answer{#1}\node[#4] (lab\thecsvrow) at (#3) {#2} node[#5] (box\thecsvrow)
1115 \newcommand\AMCrep@tikzmat[5]{\AMC@mem@answer{#1}\node[#5] (box\thecsvrow) at (#3) {#1} node[#4] (lab\thecsvrow)
1116 \newcommand\AMCrep@itemize[2]{\AMC@mem@answer{#1}\item[1] #2}
1117 \newlength\AMChorizAnswerSep
1118 \setlength{\AMChorizAnswerSep}{3em plus 4em}
1119 \newlength\AMChorizBoxSep
1120 \setlength{\AMChorizBoxSep}{1em}
1121 \newcommand\AMCrep@ligne[2]{\AMC@mem@answer{#1}%
1122 \ifAMC@textPos%
1123 \mbox{\AMC@tracebox{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}{#1}\hspace*\AMChorizBoxSep}%
1124 \else%
1125 \mbox{#1\hspace*\AMChorizBoxSep}#2}%
1126 \fi\hspace{\AMChorizAnswerSep}}
1127 \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
`\AMCendAnswer` once before answers, `\AMCanswer{⟨box⟩}{⟨text⟩}` is called for each answer (`⟨box⟩` being the box
`\AMCanswer` to be ticked and `⟨text⟩` the text associated with the proposed answer), and `\AMCendAnswer` is
called after all answers.

```

1128 \def\AMCbeginAnswer{}
1129 \def\AMCanswer#1#2{#1 #2}
1130 \def\AMCendAnswer{}

```

`\answer` The commands `\correctchoice` and `\wrongchoice` are used inside choices-like environments
`\correctchoice`
`\wrongchoice`

to give the proposed answers and specify if they are to be ticked by the students or not.

```

1131 \newcommand{\next@Answer}{%
1132   \global\advance\AMCrep@count\@ne\relax%
1133   \global\advance\AMCrep@nn\@ne\relax%
1134   \ifnum\AMCrep@nn>\AMCrep@nnmax%
1135     \global\AMCrep@nnmax=\AMCrep@nn%
1136   \fi%
1137 }
1138 \newcommand{\correctchoice}[2] []{%
1139   \next@Answer%
1140   \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%
1141   \global\AMC@bonnettrue%
1142   \AMCload@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkbox}%
1143     \else\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}\ignorespaces}
1144 \newcommand{\wrongchoice}[2] []{%
1145   \next@Answer%
1146   \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:M}\fi%
1147   \AMCload@reponse{\une@rep{\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}%
1148     \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).

```

1149 \newcommand{\AMCemptybox}[3]{%
1150   \sbox0{\wd0=#1\ht0=#2\dp0=#3\relax\box0}}
1151 \newlength\AMC@mn@test
1152 \newlength\AMC@mn@sep\AMC@mn@sep=4mm
1153 \newlength\AMC@mn@leftmargin
1154 \newlength\AMC@mn@rightmargin
1155 \newcommand\AMCmarginNote[1]{%
1156   \begin{tikzpicture}[remember picture,overlay]%
1157     \coordinate (here) at (0,0);%
1158     \pgfextractx{\AMC@mn@test}{\pgfpointdiff{\pgfpointorigin}%
1159       {\pgfpointanchor{current page}{center}}}%
1160     \ifodd\thepage%
1161       \AMC@mn@leftmargin=\oddsidemargin%
1162       \AMC@mn@rightmargin=\evensidemargin%
1163     \else
1164       \AMC@mn@leftmargin=\evensidemargin%
1165       \AMC@mn@rightmargin=\oddsidemargin%
1166     \fi
1167     \ifdim\AMC@mn@test < 1cm%
1168       \draw (current page.east |- here)+(-\AMC@mn@rightmargin-1in+\AMC@mn@sep,0pt) node[anchor=text,align=left]{#1};
1169     \else%
1170       \draw (current page.west |- here)+(0cm,0pt) node[anchor=text,align=right,text width=\AMC@mn@leftmargin]{#1};
1171     \fi%

```

```

1172 \end{tikzpicture}%
1173 }

```

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)

```

1174 \newcommand{\AMC@sz@box}{\AMCemptybox{\AMC@sz@width}{\AMC@sz@height}{\AMC@sz@depth}}
1175 %
1176 \newcommand{\AMC@sz@callin@question}{\AMCscoreZone{\AMC@sz@box}}
1177 %
1178 \newcommand{\AMC@sz@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sz@box}}}
1179 %
1180 \newcommand{\AMC@sz@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper margins}}
1181 \newcommand{\AMC@sz@callout@margins}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}

```

Let us now set up options handling.

```

1182 \newlength\AMC@sz@width
1183 \newlength\AMC@sz@height
1184 \newlength\AMC@sz@depth
1185 \def\AMC@sz@callout{}
1186 \def\AMC@sz@callin{}
1187 \define@key{AMCsz}{width}{\AMC@sz@width=#1}
1188 \define@key{AMCsz}{height}{\AMC@sz@height=#1}
1189 \define@key{AMCsz}{depth}{\AMC@sz@depth=#1}
1190 \define@key{AMCsz}{calloutside}{\def\AMC@sz@callout{#1}}
1191 \define@key{AMCsz}{callinside}{\def\AMC@sz@callin{#1}}
1192 \define@choicekey{AMCsz}{position}{none,question,margin,margins}{%
1193   \ifcsname AMC@sz@callout@#1\endcsname%
1194     \def\AMC@sz@callout{\AMC@sz@callout@#1}%
1195   \else%
1196     \def\AMC@sz@callout{}%
1197   \fi%
1198   \ifcsname AMC@sz@callin@#1\endcsname%
1199     \def\AMC@sz@callin{\AMC@sz@callin@#1}%
1200   \else%
1201     \def\AMC@sz@callin{}%
1202   \fi%
1203   \ifcsname AMC@sz@init@#1\endcsname%
1204     \csname AMC@sz@init@#1\endcsname%
1205   \fi%
1206 }
1207 \newcommand{\AMCsetScoreZone}[1]{\setkeys{AMCsz}{#1}}
1208 \AMCsetScoreZone{width=1.5em,height=1.5ex,depth=.5ex,position=none}

```

And do the same for `\AMCsetScoreZoneAnswerSheet...`

```

1209 \newcommand{\AMC@sza@box}{\AMCemptybox{\AMC@sza@width}{\AMC@sza@height}{\AMC@sza@depth}}
1210 %
1211 \newcommand{\AMC@sza@init@none}{}

```



```

1212 \newcommand{\AMC@sza@callout@none}{}
1213 \newcommand{\AMC@sza@callin@none}{}
1214 %
1215 \newcommand{\AMC@sza@init@question}{}
1216 \newcommand{\AMC@sza@callout@question}{}
1217 \newcommand{\AMC@sza@callin@question}{\AMCscoreZone{\AMC@sza@box}}
1218 %
1219 \newcommand{\AMC@sza@init@margin}{}
1220 \newcommand{\AMC@sza@callout@margin}{\hspace{Opt}\marginpar{\AMCscoreZone{\AMC@sza@box}}}
1221 \newcommand{\AMC@sza@callin@margin}{}
1222 %
1223 \newcommand{\AMC@sza@init@margin}{\PackageWarning{automultiplechoice}{Please run twice to get proper ma}}
1224 \newcommand{\AMC@sza@callout@margin}{\hspace{Opt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}
1225 \newcommand{\AMC@sza@callin@margin}{}
1226 %
1227 \newlength\AMC@sza@width
1228 \newlength\AMC@sza@height
1229 \newlength\AMC@sza@depth
1230 \def\AMC@sza@callout{}
1231 \def\AMC@sza@callin{}
1232 \define@key{AMCsza}{width}{\AMC@sza@width=#1}
1233 \define@key{AMCsza}{height}{\AMC@sza@height=#1}
1234 \define@key{AMCsza}{depth}{\AMC@sza@depth=#1}
1235 \define@key{AMCsza}{calloutsideside}{\def\AMC@sza@callout{#1}}
1236 \define@key{AMCsza}{callinsideside}{\def\AMC@sza@callin{#1}}
1237 \define@choicekey{AMCsza}{position}{none,question,margin,margins}{%
1238   \ifcsname AMC@sza@callout@#1\endcsname%
1239     \def\AMC@sza@callout{\AMC@sza@callout@#1}%
1240   \else%
1241     \def\AMC@sza@callout{}%
1242   \fi%
1243   \ifcsname AMC@sza@callin@#1\endcsname%
1244     \def\AMC@sza@callin{\AMC@sza@callin@#1}%
1245   \else%
1246     \def\AMC@sza@callin{}%
1247   \fi%
1248   \ifcsname AMC@sza@init@#1\endcsname%
1249     \csname AMC@sza@init@#1\endcsname%
1250   \fi%
1251 }
1252 \newcommand{\AMCsetScoreZoneAnswerSheet}[1]{\setkeys{AMCsza}{#1}}
1253 \AMCsetScoreZoneAnswerSheet{width=1.5em,height=1.5ex,depth=.5ex,position=none}
1254 \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.

```

1255 \newcounter{AMCquestionaff}
1256 \newcommand{\AMCnumero}[1]{\setcounter{AMCquestionaff}{#1}\addtocounter{AMCquestionaff}{-1}}

```

```

1257 \AtBeginDocument{%
1258   \ifx\@skiphyperreftrue\@undefined%
1259     \expandafter\newif\csname if@skiphyperref\endcsname%
1260   \fi%
1261 }
1262 \newcommand\AMC@stepQuestion{\ifAMCquestionNumber\@skiphyperreftrue\refstepcounter{AMCquestionaff}\@skip
1263 \newcommand\AMC@qaff{\arabic{AMCquestionaff}}

```

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

```

1264 \def\AMCbeforeQuestion{\ifAMC@qbloc\else\par\noindent\fi}
1265 \def\AMCbeginQuestion#1#2{\noindent\AMC@loc@q{#1}{#2}%
1266   \ifx\@empty\AMC@sz@callin\@empty\hspace*{1em}\fi%
1267 }
1268 \def\multiSymbole{$\clubsuit$}

```

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

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

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

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

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

```

1269 \ifx\question\undefined\else\let\question\undefined\fi
1270 \def\AMCnobloc{\AMC@qblocfalse}
1271 \def\AMCbloc{\AMC@qbloctrue}
1272 \newcommand\AMCstartWithQuestion[1]{%
1273   \global\def\AMCid@name{#1}\AMC@affecte{#1}{\AMCid@quest}%
1274   \ifAMC@calibration%
1275     \AMCmessage{Q=\the\AMCid@quest}%
1276     \immediate\write\AMC@XYFILE{\string\question{\the\AMCid@quest}{\AMCid@name}}%
1277   \fi%
1278   \start@Answers%
1279 }
1280 \newcommand\AMCexternalQuestion[2]{%
1281   \AMCstartWithQuestion{#1}%
1282   \ifAMC@calibration%
1283     \AMCmessage{B=MAX=#2}%
1284     \AMCmessage{MULT}%
1285     \AMCmessage{FQ}%
1286   \fi%
1287 }
1288 \newenvironment{question}[2][ ]{%
1289   \def\AMCcurrentenv{question}%
1290   \AMC@stepQuestion%
1291   \AMCstartWithQuestion{#2}%

```

```

1292 \AMCbeforeQuestion%
1293 \ifx\@empty\AMC@sz@callout\@empty\else%
1294 \csname\AMC@sz@callout\endcsname%
1295 \fi%
1296 \AMCtype@multifalse\ifAMC@qbloc\ifAMC@textPos\ vbox\bgroup\AMC@tracepos{1}{qtext:#2:\the\AMCid@quest,0}
1297 \ifAMC@affichekeys\index{\texttt{#2}}\ifAMC@keyslines[\texttt{#2}]\newline\fi\fi%
1298 \AMCbeginQuestion{\ifAMC@affichekeys\ifAMC@ensemble\AMC@qaff\ \fi\ifAMC@keyslines\else[\texttt{#2}]\fi}
1299 \ifAMC@calibration\immediate\write\AMC@XYFILE{\string\questionnum{\the\AMCid@etud}{\the\AMCid@quest}}
1300 \ifx\@empty\AMC@sz@callin\@empty\else%
1301 \csname\AMC@sz@callin\endcsname%
1302 \fi%
1303 \AMCformulaire@dedansfalse\setcounter{AMC@ncase}{0}%
1304 \AMC@mem@openQuestion}%
1305 {\ifAMC@qbloc\end{minipage}\ifAMC@textPos\AMC@tracepos{1}{qtext:\AMCid@name:\the\AMCid@quest,0}\egroup\AMC@
1306 \newenvironment{questionmult}[1]{%
1307 \AMC@bonnefalse\begin{question}[{\multiSymbole}]{#1}%
1308 \AMCtype@multitrue\ifAMC@calibration%
1309 \AMCmessage{MULT}\fi}%
1310 {\end{question}}}
1311 \newif\ifSurveySingleAnswer\SurveySingleAnswerfalse
1312 \newenvironment{variable-single}[2]
1313 {\def\AMCbeginQuestion##1##2{
1314 \begin{question}{#1}\scoring{v=#2}\QuestionIndicative\ifSurveySingleAnswer\else\AMCmessage{MULT}\fi
1315 \begin{tikz-single}[o]}
1316 {\end{tikz-single}
1317 \end{question}}}
1318 \newenvironment{variable-multi}[4]
1319 {\def\AMCbeginQuestion##1##2{
1320 \begin{question}{#1}\scoring{v=#4}\QuestionIndicative\ifSurveySingleAnswer\else\AMCmessage{MULT}\fi
1321 \begin{tikz-multi}[o]}
1322 \node[#3] (var) at (0,0) {#2};}
1323 {\end{tikz-multi}
1324 \end{question}}}
1325 \newenvironment{questionmultx}[1]{%
1326 \begin{group}\def\multiSymbole{\begin{questionmult}{#1}}%
1327 {\end{questionmult}\end{group}}
1328 \newdimen\ouverte@vs
1329 \newenvironment{questionouverte}[1][3cm]{%
1330 \AMC@stepQuestion%
1331 \AMCtype@multifalse\ouverte@vs=#1%
1332 \ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1333 \AMCbeginQuestion{\AMC@qaff}{}}%
1334 {\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.

```

1335 \newcommand{\explain}[1]{%
1336 \ifAMC@correc%
1337 \AMCif@env{question}{\par\noindent{\AMC@loc@explain #1}}{\AMC@error@explain}\vspace{1ex}%
1338 \else%

```

```

1339 \AMCif@env{question}{-}{\AMC@error@explain}%
1340 \fi%
1341 }
1342 \newcommand{\explaincontext}[1]{%
1343 \ifAMC@correc%
1344 #1%
1345 \fi%
1346 }

```

4.12 Scoring

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

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

```

1347 \def\scoring#1{\ifAMC@calibration\AMCmessage{B=#1}\fi\ignorespaces}
1348 \def\scoringDefaultS#1{\ifAMC@calibration\AMCmessage{BDS=#1}\fi}
1349 \def\scoringDefaultM#1{\ifAMC@calibration\AMCmessage{BDM=#1}\fi}
1350 \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], \dots, <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).

	0		0		0		0
	1		1		1		1
A	2		2		2		2
B	3		3		3		3
C	4		4		4		4
D	5		5		5		5

	0		0		0		0
	1		1		1		1
A	2		2		2		2
B	3		3		3		3
C	4		4		4		4
D	5		5		5		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}`.

A	B	C	D	E	F				
0	1	2	3	4	5	6	7	8	9
0	1	2	3	4	5	6	7	8	9

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

These two commands supports the following options (given as a comma-separated list optional argument $\langle \textit{opts} \rangle$):

- `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.

```

1351 \newcount\AMC@chiffres
1352 \newdimen\AMCcodeHspace\AMCcodeHspace=.5em
1353 \newdimen\AMCcodeVspace\AMCcodeVspace=.5em
1354 \newcommand\AMCcodeID@squarebrackets[2]{#1[#2]}
1355 \newcommand\AMCcodeID@dot[2]{#1.#2}
1356 \newcommand\AMCcodeID@@[1]{%
1357   \expandafter\def\expandafter\AMCcodeID\expandafter{\csname AMCcodeID@#1\endcsname}%
1358   \def\AMCcodeID@mode{#1}%
1359 }
1360 \AMCcodeID@@{squarebrackets}
1361 \ExplSyntaxOn
1362
1363 \clist_new:N \amc_code_descr_clist
1364 \seq_new:N \amc_code_digits_seq
1365 \int_new:N \amc_code_digit_n_int
1366 \bool_new:N \amc_code_vertical_bool
1367 \bool_new:N \amc_code_top_bool
1368 \bool_new:N \amc_code_multi_bool
1369 \clist_new:N \amc__multi_clist
1370
1371 \cs_new:Npn \amc_code_init:N #1 {
1372   \def\AMCbeginQuestion##1##2{}
1373   \def\AMCbeforeQuestion{}
1374   \AMCnoScoreZone
1375   \AMCquestionNumberfalse
1376   \setlength{\parindent}{0pt}
1377   \AMCnobloc
1378   \int_set:Nn \amc_code_digit_n_int { \clist_count:N #1 }
1379 }
1380
1381 \cs_new:Nn \amc_code_digit_init: {
1382   \QuestionIndicative
1383   \global\AMCrep@count=\z@
1384 }
1385
1386 \cs_new:Npn \amc_code_digit:n #1 {
1387   \global\advance\AMCrep@count\@ne\relax
1388   \ifAMC@calibration\AMCmessage{ REP = \the\AMCrep@count : M }\fi
1389   \hbox{\AMC@keyBox@{#1}{1}{case : \AMCid@name : \the\AMCid@quest , \the\AMCrep@count}}
1390   \bool_if:NTF \amc_code_vertical_bool {

```

```

1391 \vspace{\AMCcodeVspace}
1392 }{
1393 \hspace{\AMCcodeHspace}
1394 }
1395 }
1396
1397 \keys_define:nn { amccode } {
1398   vertical .bool_set:N = \amc_code_vertical_bool,
1399   vertical .initial:n = { true },
1400   vertical .default:n = { true },
1401   v .code:n = { \bool_set_true:N \amc_code_vertical_bool },
1402   h .code:n = { \bool_set_false:N \amc_code_vertical_bool },
1403   top .bool_set:N = \amc_code_top_bool,
1404   top .initial:n = { false },
1405   top .default:n = { true },
1406   multi .bool_set:N = \amc_code_multi_bool,
1407   multi .initial:n = { false },
1408   multi .default:n = { true }
1409 }
1410
1411 \cs_new_nopar:Nn \amc_multi_report: {
1412   \ifAMC@calibration
1413     \immediate\write\AMC@XYFILE{\string\with{multi=\clist_use:Nn\amc__multi_clist{,}}}
1414   \fi
1415 }
1416 \cs_new_eq:NN \AMC@multi@report \amc_multi_report:
1417 \int_new:N \amc_multi_count_int
1418 \cs_new_nopar:Nn \amc_multi_clear: {
1419   \int_gzero:N \amc_multi_count_int
1420 }
1421 \cs_new_eq:NN \AMC@multiclear \amc_multi_clear:
1422
1423 \cs_new:Npn \amc_code_generate:nNn #1#2#3 {
1424   { \keys_set:nn { amccode } { #3 }
1425     \bool_if:NTF \amc_code_multi_bool {
1426       \clist_gset:Nn \amc__multi_clist { #1 }
1427     } {}
1428     \bool_if:NTF \amc_code_multi_bool { \int_gincr:N \amc_multi_count_int } {}
1429     \amc_code_init:N #2
1430     \clist_map_inline:Nn #2 { % iterates over 'digits'
1431       \begin{question}{
1432         \AMCcodeID{ #1 \bool_if:NTF
1433           \amc_code_multi_bool
1434           { * \int_use:N \amc_multi_count_int } {} }
1435         { \int_use:N \amc_code_digit_n_int }
1436       }
1437       \amc_code_digit_init:
1438       \seq_set_split:Nnn \amc_code_digits_seq {} { ##1 }
1439       \bool_if:NTF \amc_code_vertical_bool {
1440         \hspace{0pt}
1441         \bool_if:NTF \amc_code_top_bool { \vtop } { \vbox }
1442         \bgroup
1443       }{

```

```

1444     \hbox\bgroup
1445   }
1446   \seq_map_inline:Nn \amc_code_digits_seq {
1447     % iterates over available characters for 'digit'
1448     \amc_code_digit:n { ####1 }
1449   }
1450   \bool_if:NTF \amc_code_vertical_bool {
1451     \vspace{-\AMCcodeVspace}\egroup
1452     \hspace{\AMCcodeHspace}
1453   }{
1454     \egroup\vspace{\AMCcodeVspace}
1455     \par
1456   }
1457   \end{question}
1458   \int_decr:N \amc_code_digit_n_int
1459 }
1460 }
1461 }
1462
1463 \cs_new:Npn \amc_code_generate:nnn #1#2#3 {
1464   \clist_set:Nn \amc_code_descr_clist { #2 }
1465   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1466 }
1467 \cs_generate_variant:Nn \amc_code_generate:nnn { xxx }
1468 \newcommand{\AMCcodeGrid}[3][ ]{
1469   \amc_code_generate:xxx { #2 } { #3 } { #1 }
1470 }
1471
1472 \cs_new:Npn \amc_code_generate_integer:nnn #1#2#3 {
1473   \clist_clear:N \amc_code_descr_clist
1474   \prg_replicate:nn { #2 } { \clist_put_right:Nn \amc_code_descr_clist { 0123456789 } }
1475   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1476 }
1477 \cs_generate_variant:Nn \amc_code_generate_integer:nnn { xxx }
1478 \newcommand{\AMCcodeGridInt}[3][ ]{
1479   \amc_code_generate_integer:xxx { #2 } { #3 } { #1 }
1480 }
1481
1482 \cs_new:Npn \amc_code_generate_integer_v:nn #1#2 {
1483   \amc_code_generate_integer:nnn { #1 } { #2 } { v }
1484 }
1485 \cs_new:Npn \amc_code_generate_integer_h:nn #1#2 {
1486   \amc_code_generate_integer:nnn { #1 } { #2 } { h }
1487 }
1488 \cs_generate_variant:Nn \amc_code_generate_integer_v:nn { xx }
1489 \cs_generate_variant:Nn \amc_code_generate_integer_h:nn { xx }
1490 \cs_new_eq:NN \AMCcode \amc_code_generate_integer_v:xx
1491 \cs_new_eq:NN \AMCcodeH \amc_code_generate_integer_h:xx
1492
1493 \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 ?																				
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;"><input type="checkbox"/>0</td> <td style="padding: 2px 5px;"><input type="checkbox"/>1</td> <td style="padding: 2px 5px;"><input type="checkbox"/>2</td> <td style="padding: 2px 5px;"><input checked="" type="checkbox"/>3</td> <td style="padding: 2px 5px;"><input type="checkbox"/>4</td> <td style="padding: 2px 5px;"><input type="checkbox"/>5</td> <td style="padding: 2px 5px;"><input type="checkbox"/>6</td> <td style="padding: 2px 5px;"><input type="checkbox"/>7</td> <td style="padding: 2px 5px;"><input type="checkbox"/>8</td> <td style="padding: 2px 5px;"><input type="checkbox"/>9</td> </tr> <tr> <td style="padding: 2px 5px;"><input type="checkbox"/>0</td> <td style="padding: 2px 5px;"><input type="checkbox"/>1</td> <td style="padding: 2px 5px;"><input type="checkbox"/>2</td> <td style="padding: 2px 5px;"><input type="checkbox"/>3</td> <td style="padding: 2px 5px;"><input type="checkbox"/>4</td> <td style="padding: 2px 5px;"><input checked="" type="checkbox"/>5</td> <td style="padding: 2px 5px;"><input type="checkbox"/>6</td> <td style="padding: 2px 5px;"><input type="checkbox"/>7</td> <td style="padding: 2px 5px;"><input type="checkbox"/>8</td> <td style="padding: 2px 5px;"><input type="checkbox"/>9</td> </tr> </table> </div>	<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` option is used and if we are currently in the question part (not in the answer sheet), or `<showcommand>{<arg1>}{<arg2>}` when all the boxes are to be produced.

```
1494 \newcommand\AMCformatChoices[4]{%
1495   \global\AMCrep@count=\z@%
1496   \AMC@if@separate@question{%
1497     \AMC@mem@add{\global\AMCrep@count=\z@%
1498       #1{#3}{#4}}%
1499   }%
1500   \ifAMC@ensemble%
1501     #2{#3}{#4}%
1502     \AMCmessage{QPART}%
1503   \else%
1504     #1{#3}{#4}%
1505   \fi%
1506 }
```

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

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

```
1507 \ExplSyntaxOn
1508
1509 \cs_generate_variant:Nn \tl_replace_once:Nnn { Nxn }
1510
1511 \tl_new:N \amc_ee_tl
1512 \seq_new:N \amc_ee_seq
```


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

```

1513 \group_begin:
1514 \char_set_catcode_other:N E
1515 \tex_lowercase:D
1516 {
1517   \cs_new:Npn \amc_read_scientific:NNn #1 #2 #3 {
1518     \tl_set:Nn \amc_ee_tl { #3 }
1519     \tl_replace_once:Nxn \amc_ee_tl { E } { e }
1520     \seq_set_split:NnV \amc_ee_seq e \amc_ee_tl
1521     \fp_set:Nn #1 { \seq_item:Nn \amc_ee_seq 1 }
1522     \int_set:Nn #2 { \seq_item:Nn \amc_ee_seq 2 }
1523   }
1524 }
1525 \group_end:
1526
1527 \cs_generate_variant:Nn \amc_read_scientific:NNn { NNf, NNx }
1528
1529 \fp_new:N \amc_fulls_fp
1530 \cs_new:Npn \amc_fp_decompose:NNn #1 #2 #3 {
1531   \fp_set:Nn \amc_fulls_fp { #3 }

```

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

```

1532   \amc_read_scientific:NNx #1 #2
1533   { \fp_to_scientific:N \amc_fulls_fp e 0 }
1534 }
1535 \cs_generate_variant:Nn \amc_fp_decompose:NNn { NNx }
1536
1537 \fp_new:N \amc_num_mantissa_fp
1538 \int_new:N \amc_num_exponent_int
1539 \cs_new:Npn \amc_fp_n_significant_digits:Nnn #1 #2 #3 {
1540   \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_exponent_int
1541   { #2 }
1542   \fp_set:Nn #1
1543   { round(\amc_num_mantissa_fp * 10^((#3)-1)) }
1544   \fp_compare:nTF { abs(#1) >= 10^(#3) }
1545   {
1546     \fp_set:Nn #1 { #1 / 10 }
1547   } { }
1548 }
1549
1550 \fp_new:N \amc_num_nsig_fp
1551 \cs_new:Npn \amc_fp_show_n_significant_digits:nn #1 #2 {
1552   \amc_fp_n_significant_digits:Nnn \amc_num_nsig_fp { #1 } { #2 }
1553 }
1554 \cs_new_eq:NN \AMCsignificantDigits \amc_fp_show_n_significant_digits:nn
1555
1556 \cs_new:Npn \amc_fp_show_significant_digits: {
1557   \fp_use:N \amc_num_nsig_fp
1558 }
1559 \cs_new_eq:NN \AMCshowSignificantDigits \amc_fp_show_significant_digits:
1560
1561 \cs_new:Npn \amc_fp_n_digits:Nnn #1 #2 #3 {

```

```

1562 \fp_set:Nn #1
1563 { round((#2) * \amc_num_base_int^(#3)) }
1564 }
1565
1566 \int_new:N \amc_todigits_int
1567 \cs_new:Npn \amc_fp_to_digits:Nnnn #1 #2 #3 #4 {
1568   \clist_clear:N #1
1569   \int_set:Nn \amc_todigits_int { \fp_eval:n { abs(round(#2)) } }
1570   \prg_replicate:nn { #3 } {
1571     \clist_put_left:Nx #1 { \int_mod:nn \amc_todigits_int { #4 } }
1572     \int_set:Nn \amc_todigits_int
1573     { \int_div_truncate:nn \amc_todigits_int { #4 } }
1574   }
1575   \int_compare:nNnTF \amc_todigits_int = 0 { } {
1576     \message{^^J!~Error:~number~too~large,
1577       ~some~digits~will~be~discarded^^J}
1578   }
1579 }
1580
1581 \cs_new:Npn \amc_invalid_digits:Nn #1 #2 {
1582   \clist_clear:N #1
1583   \prg_replicate:nn { #2 } { \clist_put_left:Nx #1 { -1 } }
1584 }
1585
1586 \cs_new:Npn \amc_get_fp_sign:Nn #1 #2 {
1587   \fp_compare:nNnTF #2 < 0 {
1588     \int_set:Nn #1 { -1 }
1589   }{
1590     \fp_compare:nNnTF #2 > 0 {
1591       \int_set:Nn #1 { 1 }
1592     }{
1593       \int_set:Nn #1 { 0 }
1594     }
1595   }
1596 }
1597
1598 \cs_new:Npn \amc_get_int_sign:Nn #1 #2 {
1599   \int_compare:nNnTF #2 < 0 {
1600     \int_set:Nn #1 { -1 }
1601   }{
1602     \int_compare:nNnTF #2 > 0 {
1603       \int_set:Nn #1 { 1 }
1604     }{
1605       \int_set:Nn #1 { 0 }
1606     }
1607   }
1608 }
1609
1610 \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:

```

1611 \def\AMCnTextGoto{}
1612 \def\AMCnTextVHead#1{\emph{b#1}}
1613 \newdimen\AMCnumeric@Hspace\AMCnumeric@Hspace=.5em
1614 \newdimen\AMCnumeric@Vspace\AMCnumeric@Vspace=1ex
1615 \ExplSyntaxOn
1616
1617 \keys_define:nn { amcnumeric } {
1618   Tsign .code:n = {\def\AMCnTextSign{#1}},
1619   Tsign .initial:n = {},
1620   Tpoint .code:n = {\def\AMCdecimalPoint{#1}},
1621   Tpoint .initial:n = { \raisebox{1ex}{\bf .} },
1622   Texponent .code:n = {\def\AMCexponent{#1}},
1623   Texponent .initial:n = { $\times 10$\textasciicircum },
1624   vspace .code:n = {\AMCnumeric@Vspace=#1},
1625   hspace .code:n = {\AMCnumeric@Hspace=#1},
1626   bordercol .code:n = {\def\AMCncol@Border{#1}},
1627   bordercol .initial:n = { lightgray },
1628   borderwidth .code:n = {\def\AMCncol@BorderWidth{#1}},
1629   borderwidth .initial:n = { 1mm },
1630   backgroundcol .code:n = {\def\AMCncol@Background{#1}},
1631   backgroundcol .initial:n = { white },
1632   digits .int_set:N = \amc_num_ndigits_int,
1633   digits .initial:n = { 3 },
1634   decimals .int_set:N = \amc_num_decd_int,
1635   decimals .initial:n = { 0 },
1636   exponent .int_set:N = \amc_num_expo_int,
1637   exponent .initial:n = { 0 },
1638   base .int_set:N = \amc_num_base_int,
1639   base .initial:n = { 10 },
1640   sign .bool_set:N = \amc_num_sign_bool,
1641   sign .initial:n = { true },
1642   sign .default:n = { true },
1643   exposign .bool_set:N = \amc_num_exposign_bool,
1644   exposign .initial:n = { true },
1645   exposign .default:n = { true },
1646   strict .bool_set:N = \amc_num_strict_bool,
1647   strict .initial:n = { false },
1648   strict .default:n = { true },
1649   scoring .bool_set:N = \amc_num_scoring_bool,
1650   scoring .initial:n = { true },
1651   scoring .default:n = { true },
1652   ignoreblank .bool_set:N = \amc_num_ignoreblank_bool,
1653   ignoreblank .initial:n = { false },
1654   ignoreblank .default:n = { true },
1655   vertical .bool_set:N = \amc_num_vertical_bool,
1656   vertical .initial:n = { false },
1657   vertical .default:n = { true },
1658   expoververtical .bool_set:N = \amc_num_expoververtical_bool,
1659   expoververtical .initial:n = { false },
1660   expoververtical .default:n = { true },
1661   reverse .bool_set:N = \amc_num_reverse_bool,
1662   reverse .initial:n = { false },

```

```

1663 reverse .default:n = { true },
1664 vhead .bool_set:N = \amc_num_vhead_bool,
1665 vhead .initial:n = { false },
1666 vhead .default:n = { true },
1667 Tvhead .code:n = {\clist_set:Nx \amc_tvhead_clist {\clist_reverse:n {#1}}},
1668 Tvhead .initial:n = {}, % \c_empty_clist does not work with \clist_reverse:n
1669 vheadunitindex .int_set:N = \amc_vheadunitindex_int,
1670 vheadunitindex .initial:n = 0,
1671 nozero .bool_set:N = \amc_num_nozero_bool,
1672 nozero .initial:n = { false },
1673 nozero .default:n = { true },
1674 significant .bool_set:N = \amc_num_significant_bool,
1675 significant .initial:n = { false },
1676 significant .default:n = { true },
1677 scoreexact .code:n = {\def\AMC@numeric@scoreexact{#1}},
1678 scoreexact .initial:n = { 2 },
1679 scoreapprox .code:n = {\def\AMC@numeric@scoreapprox{#1}},
1680 scoreapprox .initial:n = { 1 },
1681 scorewrong .code:n = {\def\AMC@numeric@scorewrong{#1}},
1682 scorewrong .initial:n = { 0 },
1683 exact .int_set:N = \amc_num_exact_int,
1684 exact .initial:n = { 0 },
1685 approx .int_set:N = \amc_num_approx_int,
1686 approx .initial:n = { 0 },
1687 keepas .code:n = {\def\AMC@numeric@keepas{#1}},
1688 keepas .initial:n = {},
1689 alsocorrect .code:n = {\def\AMC@numeric@alsocorrect{#1}},
1690 alsocorrect .initial:n = {}
1691 }
1692
1693 \cs_new:Npn \amc_num_setopts #1 {
1694   \keys_set:nn { amcnumeric } { #1 }
1695 }
1696
1697 \cs_new_nopar:Nn \amc_num_check_score_opts: {
1698   \bool_if:NTF \amc_num_ignoreblank_bool {
1699     \int_compare:nNnTF \amc_num_base_int = { 10 } { } {
1700       \message{^^J!~Error:~ignoreblank~can~only~be~used~with~number~base~10^^J}
1701     }
1702   } {}
1703 }
1704
1705 \cs_new_eq:NN \AMCnumericOpts \amc_num_setopts
1706

```

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

```

1707 \cs_new:Npn \amc_num_char:nn #1 #2 {
1708   \global\advance\AMCrep@count\@ne\relax
1709   \AMCmessage{REP= \the\AMCrep@count :
1710     \ifx#2\AMC@checkbox B\else M\fi }
1711   \ifAMC@correc
1712     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :

```

```

1713     \the\AMCid@quest , \the\AMCrep@count}
1714   \else
1715     \protect\AMC@keyBox@{#1}{\1}{case : \AMCid@name :
1716     \the\AMCid@quest , \the\AMCrep@count}
1717   \fi
1718 }

```

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

```

1719 \int_new:N \amc_num_digit_value_int
1720 \tl_new:N \amc_num_digit_value_tl
1721 \cs_new:Npn \amc_num_digit_box:nn #1 #2 {
1722   \int_set:Nn \amc_num_digit_value_int { #1 }
1723   \tl_set:Nn \amc_num_digit_value_tl {
1724     \int_compare:nNnTF { \amc_num_digit_value_int } < { 10 }
1725     { \int_to_arabic:n { \amc_num_digit_value_int } }
1726     { \int_to_Alph:n { \amc_num_digit_value_int - 9 } }
1727   }
1728   \int_compare:nNnTF { #1 } = { #2 } {
1729     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1730     {\AMC@checkbox}
1731   } {
1732     \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1733     {}
1734   }
1735 }

```

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 $\langle negative \rangle$).

```

1736 \cs_new:Npn \amc_num_sign_boxes:Nn #1 #2 {
1737   \int_case:nn { #1 } {
1738     { -1 } {
1739       \hbox{\amc_num_char:nn{+}{}}
1740       \vspace{\AMCnumeric@Vspace}
1741       \AMCmessage{B=set. sign #2 =1}
1742       \hbox{\amc_num_char:nn{-}{\AMC@checkbox}}
1743       \AMCmessage{B=set. sign #2 =-1}
1744     }
1745     { 1 } {
1746       \hbox{\amc_num_char:nn{+}{\AMC@checkbox}}
1747       \vspace{\AMCnumeric@Vspace}
1748       \AMCmessage{B=set. sign #2 =1}
1749       \hbox{\amc_num_char:nn{-}{}}
1750       \AMCmessage{B=set. sign #2 =-1}
1751     }
1752     { 0 } {
1753       \hbox{\amc_num_char:nn{+}{}}
1754       \vspace{\AMCnumeric@Vspace}
1755       \AMCmessage{B=set. sign #2 =1}
1756       \hbox{\amc_num_char:nn{-}{\AMC@checkbox}}
1757       \AMCmessage{B=set. sign #2 =-1}
1758     }
1759   }

```

1760 }

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

```

1761 \cs_new:Npn \amc_num_digit_boxes_h:nnn #1 #2 #3 {
1762   \int_step_inline:nnnn
1763   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1764   { 1 } { #3 - 1 } {
1765     \amc_num_digit_box:nn { ##1 }{ #2 }
1766     \AMCmessage{B= set. #1 = ##1}
1767     \hspace{\AMCnumeric@Hspace}
1768   }
1769   \hspace{-\AMCnumeric@Hspace}
1770 }
1771
1772 \cs_new:Npn \amc_num_digit_boxes_v:nnn #1 #2 #3 {
1773   \int_step_inline:nnnn
1774   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1775   { 1 } { #3 - 1 } {
1776     \vbox{\hbox{
1777       \amc_num_digit_box:nn { ##1 }{ #2 }
1778     }}
1779     \AMCmessage{B= set. #1 = ##1}
1780     \int_compare:nNnTF { ##1 } < { #3 - 1 } {
1781       \vspace{\AMCnumeric@Vspace}
1782     } {}
1783   }
1784 }
1785
1786 \int_new:N \amc_num_first_digit_int
1787 \cs_new:Npn \amc_num_digit_boxes_vr:nnn #1 #2 #3 {
1788   \int_set:Nn \amc_num_first_digit_int
1789   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1790   \int_step_inline:nnnn { #3 - 1 } { -1 }
1791   \amc_num_first_digit_int {
1792     \vbox{\hbox{
1793       \amc_num_digit_box:nn { ##1 }{ #2 }
1794     }}
1795     \AMCmessage{B= set. #1 = ##1}
1796     \int_compare:nNnTF { ##1 } > \amc_num_first_digit_int {
1797       \vspace{\AMCnumeric@Vspace}
1798     } {}
1799   }
1800 }

```

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

```

1801 \cs_new:Npn \amc_num_integer_boxes_v:Nnn #1 #2 #3 {
begin a loop over all digits,
1802   \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1803   \clist_map_inline:Nn #1 {

```

place the decimal point if necessary,

```
1804 \int_compare:nNnTF \amc_num_digit_int = { #3 } {
1805   \hbox{ \AMCdecimalPoint }\hspace{\AMCnumeric@Hspace}
1806 } { }
```

draw the box for this digit,

```
1807 \hbox{\vbox{
1808   \bool_if:NTF \amc_num_vhead_bool {
1809     \vbox{\hbox to \AMC@boxedwidth{\hfill\AMCnTextVHead{ \int_eval:n
1810       { \amc_num_digit_int - 1 } }\hfill}}
1811     \vspace{\AMCnumeric@Vspace}
1812   } { }
1813   \bool_if:NTF \amc_num_reverse_bool {
1814     \amc_num_digit_boxes_vr:nnn { #2
1815       \int_to_Alph:n \amc_num_digit_int }
1816     { ##1 } { \amc_num_base_int }
1817   } {
1818     \amc_num_digit_boxes_v:nnn { #2
1819       \int_to_Alph:n \amc_num_digit_int }
1820     { ##1 } { \amc_num_base_int }
1821   }
1822 }}
```

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

```
1823 \int_compare:nNnTF \amc_num_digit_int > 1 {
1824   \hspace{\AMCnumeric@Hspace}
1825 } { }
1826 \int_decr:N \amc_num_digit_int
1827 }
1828 }
1829
```

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

```
1830
1831 \cs_new:Npn \amc_num_integer_boxes_h:Nnn #1 #2 #3 {
1832   \vbox{
1833     \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1834     \clist_map_inline:Nn #1 {
1835       \int_compare:nNnTF
1836       \amc_num_digit_int = { #3 } {
1837         \hbox{ \AMCdecimalPoint }
1838       } { }
1839       \hbox{
1840         \amc_num_digit_boxes_h:nnn { #2
1841           \int_to_Alph:n \amc_num_digit_int }
1842         { ##1 } \amc_num_base_int
1843       }
1844       \int_compare:nNnTF \amc_num_digit_int > 1 {
1845         \vspace{\AMCnumeric@Vspace}
1846       } { }
1847       \int_decr:N \amc_num_digit_int
1848     }
1849   }
1850
```

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.

```

1851
1852 \cs_new:Npn \amc_num_integer_boxes:NnnNN #1 #2 #3 #4 #5 {
1853   \hbox{
1854     \bool_if:NTF { #4 } {
1855       \vbox{
1856         \ifx\AMCnTextSign\@empty\@empty\else
1857           \hbox{\AMCnTextSign}\vspace{\AMCnumeric@Vspace}\fi
1858         \bool_if:NTF \amc_num_strict_bool {
1859           \int_compare:nNnTF { #5 } = 0 {
1860             \amc_num_sign_boxes:Nn { 1 } { #2 }
1861           }{
1862             \amc_num_sign_boxes:Nn { #5 } { #2 }
1863           }
1864         }{
1865           \amc_num_sign_boxes:Nn { #5 } { #2 }
1866         }
1867       }
1868       \hspace{.5em}
1869       \vrule
1870       \hspace{.5em}
1871     } { }
1872   \hbox{
1873     \bool_if:NTF \amc_num_vertical_bool
1874     \amc_num_integer_boxes_v:Nnn \amc_num_integer_boxes_h:Nnn
1875     #1 { #2 } { #3 }
1876   }
1877 }
1878 }
1879

```

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 `<n>`-digits boxes (from which `<decimals>` are for decimals), with name prefix `<prefix>`, using a sign variable if `<sign bool>` is true.

```

1880
1881 \cs_new:Npn \amc_num_build_integer_scoring:Nnnnn #1 #2 #3 #4 #5 {
1882   \amc_num_check_score_opts:
1883   \tl_clear:N #1
1884   \int_set_eq:NN \amc_num_digit_int { #4 }
1885   \int_while_do:nNnn \amc_num_digit_int > 0 {
1886     \bool_if:NTF \amc_num_strict_bool {
1887       \AMCmessage{B=requires. #3
1888         \int_to_Alph:n \amc_num_digit_int = 1}
1889     } {
1890       \AMCmessage{B=default. #3
1891         \int_to_Alph:n \amc_num_digit_int =
1892         \bool_if:NTF \amc_num_ignoreblank_bool { } { 0 }
1893       }
1894     }
1895     \int_compare:nNnTF \amc_num_digit_int = #4 { } {
1896       \bool_if:NTF \amc_num_ignoreblank_bool {

```



```

1897     \tl_put_right:Nx #1 { ~.~ }
1898     \int_compare:nNnTF \amc_num_digit_int = #5 {
1899         \tl_put_right:Nx #1 { "." ~.~ }
1900     } { }
1901 } {
1902     \tl_put_left:Nn #1 { ( }
1903     \tl_put_right:Nx #1 { ) } *
1904     \int_use:N \amc_num_base_int + }
1905 }
1906 }
1907 \tl_put_right:Nx #1
1908 { #3 \int_to_Alph:n \amc_num_digit_int }
1909 \int_decr:N \amc_num_digit_int
1910 }
1911 \bool_if:NTF \amc_num_ignoreblank_bool {
1912     \tl_put_left:Nn #1 { ( 0 + ( }
1913     \tl_put_right:Nn #1 { ) ) }
1914     \int_compare:nNnTF \amc_num_decd_int > 0 {
1915         \tl_put_right:Nx #1 { * ( \int_use:N \amc_num_base_int ** \int_eval:n { #5 } ) }
1916     } { }
1917 } {
1918     \tl_put_left:Nn #1 { ( }
1919     \tl_put_right:Nn #1 { ) }
1920 }
1921 \bool_if:NTF { #2 } {
1922     \bool_if:NTF \amc_num_strict_bool {
1923         \AMCmessage{B=requires. sign #3 =1}
1924     } {
1925         \AMCmessage{B=default. sign #3 =1}
1926     }
1927     \tl_put_right:Nx #1 { * ( sign #3 ) }
1928 } { }
1929 }
1930

```

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

```

1931
1932 \fp_new:N \amc_num_result_fp
1933 \fp_new:N \amc_num_correct_fp
1934 \clist_new:N \amc_num_digits_clist
1935 \clist_new:N \amc_num_expo_digits_clist
1936 \int_new:N \amc_num_digit_int
1937 \int_new:N \amc_num_sign_int
1938 \int_new:N \amc_num_expo_sign_int
1939 \tl_new:N \amc_num_compute_tl
1940 \tl_new:N \amc_num_expo_tl
1941 \int_new:N \amc_num_correct_expo_int
1942
1943 \cs_new:Npn \amc_numeric_show:nn #1 #2 {

```

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

```

1944 \ifAMC@ensemble\ifAMCformulaire@dedans
1945     \AMCmessage{Q=\the\AMCid@quest}
1946 \fi\fi

```

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

```

1947 {\keys_set:nn { amcnumeric } { #2 }
if Tvhead is set, we adapt the AMCncontextVHead macro
1948 \int_compare:nTF {\amc_vheadunitindex_int = 0}
1949 {\int_set:Nn \amc_vheadunitindex_int {\amc_num_decd_int + 1}}
1950 {}
1951
1952 \clist_if_empty:NTF \amc_tvhead_clist
1953 {}
1954 {
1955   \def\AMCncontextVHead##1{
1956     \int_set:Nn \l_tmpa_int
1957     {\int_max:nn
1958      {(##1) - \amc_num_decd_int + \amc_vheadunitindex_int}
1959      {0}}
1960   \emph{\clist_item:Nn \amc_tvhead_clist {\l_tmpa_int}}}}
1961
1962   \bool_if:nTF { \bool_if_p:N\amc_num_significant_bool
1963     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1964     \message{^^J!~AMCnumeric~Error:~significant=true~can't~be~used~with~base!=10.^^J}
1965   } {}
1966   \bool_if:nTF { \int_compare_p:n { \amc_num_expo_int != 0 }
1967     && \int_compare_p:n { \amc_num_base_int != 10 } } {
1968     \message{^^J!~AMCnumeric~Error:~scientific~notation~can't~be~used~with~base!=10.^^J}
1969   } {}

```

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

```

1970   \ifx\@empty#1\@empty
1971     \fp_set:Nn \amc_num_correct_fp { 0 }
1972     \fp_set:Nn \amc_num_mantissa_fp { 0 }
1973     \int_set:Nn \amc_num_correct_expo_int { 0 }
1974   \else
1975     \bool_if:NTF \amc_num_significant_bool {
1976       \amc_fp_n_significant_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_ndigits_int
1977     } {
1978       \int_compare:nNnTF \amc_num_expo_int > 0 {
1979         \amc_fp_decompose:Nnn \amc_num_mantissa_fp \amc_num_correct_expo_int { #1 }
1980         \int_compare:nNnTF { \amc_num_ndigits_int - \amc_num_decd_int } > 1 {
1981           \fp_set:Nn \amc_num_mantissa_fp {
1982             \amc_num_mantissa_fp * \amc_num_base_int^( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1983           }
1984           \int_set:Nn \amc_num_correct_expo_int {
1985             \amc_num_correct_expo_int - ( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
1986           }
1987         } {}
1988         \amc_fp_n_digits:Nnn \amc_num_correct_fp \amc_num_mantissa_fp \amc_num_decd_int
1989       } {
1990         \amc_fp_n_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_decd_int
1991       }
1992     }
1993   \fi

```

Now extracts the required digits:

```

1994 \ifx\@empty#1\@empty
1995   \amc_invalid_digits:Nn \amc_num_digits_clist \amc_num_ndigits_int
1996   \amc_invalid_digits:Nn \amc_num_expo_digits_clist \amc_num_expo_int
1997   \int_set:Nn \amc_num_sign_int { 0 }
1998   \int_set:Nn \amc_num_expo_sign_int { 0 }
1999 \else
2000   \amc_fp_to_digits:Nnnn \amc_num_digits_clist \amc_num_correct_fp
2001   \amc_num_ndigits_int \amc_num_base_int
2002   \amc_get_fp_sign:Nn \amc_num_sign_int \amc_num_correct_fp
2003   \int_compare:nNnTF \amc_num_expo_int > 0 {
2004     \amc_fp_to_digits:Nnnn \amc_num_expo_digits_clist \amc_num_correct_expo_int
2005     \amc_num_expo_int \amc_num_base_int
2006     \amc_get_int_sign:Nn \amc_num_expo_sign_int \amc_num_correct_expo_int
2007   } {}
2008 \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.

```

2009 \fp_set:Nn \amc_num_result_fp { #1 }
2010 \AMCmessage{B=numval=\fp_to_scientific:N \amc_num_result_fp ,
2011   numex=\int_use:N \amc_num_exact_int,
2012   numapp=\int_use:N \amc_num_approx_int,
2013   numsex=\AMC@numeric@scoreexact,
2014   numsapp=\AMC@numeric@scoreapprox
2015 }
2016 \bool_if:NTF \amc_num_scoring_bool {
2017   \AMCmessage{B=haut=mz=d=undef,p=undef,
2018     formula=(Vdifference <= \int_use:N \amc_num_exact_int ?
2019     \AMC@numeric@scoreexact :
2020     \int_compare:nNnTF \amc_num_approx_int = 0 {
2021       \AMC@numeric@scorewrong
2022     } {
2023       (Vdifference <= \int_use:N \amc_num_approx_int ?
2024       \AMC@numeric@scoreapprox : \AMC@numeric@scorewrong)
2025     }
2026   }}
2027 } {}
2028 \amc_num_build_integer_scoring:Nnnnn
2029   \amc_num_compute_tl \amc_num_sign_bool { digit } \amc_num_ndigits_int
2030   \amc_num_decd_int
2031 \int_compare:nNnTF \amc_num_expo_int > 0 {
2032   \amc_num_build_integer_scoring:Nnnnn
2033   \amc_num_expo_tl \amc_num_exposign_bool { expo } \amc_num_expo_int { 0 }
2034   \AMCmessage{B= set. intE = \amc_num_expo_tl}
2035 } {}
2036 \AMCmessage{B= set.intV = \fp_to_int:N \amc_num_correct_fp ,
2037   set.intXX = \amc_num_compute_tl }
2038 \int_compare:nNnTF \amc_num_expo_int > 0 {
2039   \AMCmessage{B= set.intX = intXX * \int_use:N \amc_num_base_int ** ( intE - (\int_use:N \amc_num_correc
2040 }{
2041   \AMCmessage{B= set.intX = intXX}
2042 }
2043 \int_compare:nNnTF \amc_num_expo_int > 0 {
2044   \AMCmessage{B= set.valueX = intXX * \int_use:N \amc_num_base_int ** (intE - \int_use:N \amc_num_decd

```

```

2045   }{
2046     \AMCmessage{B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (- \int_use:N\amc_num_decd_int)}
2047   }
2048   \ifx\@empty\AMC@numeric@keepas\@empty\else
2049     \AMCmessage{B= setglobal.\AMC@numeric@keepas = valueX}
2050   \fi
2051   \ifx\@empty#1\@empty
2052     \bool_if:NTF \amc_num_significant_bool {
2053       \AMCmessage{B=set.Vdifference=0}
2054     }{
2055       \ifx\@empty\AMC@numeric@alsocorrect\@empty
2056         \AMCmessage{B=set.Vdifference=0}
2057       \else
2058         \AMCmessage{B="set.Vdifference =
2059           amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\amc_r
2060           "}
2061       \fi
2062     }
2063   \else
2064     \bool_if:NTF \amc_num_significant_bool {
2065       \AMCmessage{B=set.Vdifference="min( abs((intV)-(intX)) ,
2066         abs(\int_use:N\amc_num_base_int * (intV) - (intX)) ,
2067         abs((intV) - \int_use:N\amc_num_base_int * (intX)) )"}
2068     } {
2069       \ifx\@empty\AMC@numeric@alsocorrect\@empty
2070         \AMCmessage{B=set.Vdifference=abs((intV)-(intX))}
2071       \else
2072         \AMCmessage{B="set.Vdifference =
2073           min( amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\am
2074           abs((intV)-(intX)) )"}
2075       \fi
2076     }
2077   \fi

```

Begin now with the frame around all the boxes:

```

2078   \ifAMC@extractOnly\else
2079     \vspace{1.5ex}\par{
2080       \fboxrule=\AMCncol@BorderWidth
2081       \fcolorbox{\AMCncol@Border}{\AMCncol@Background}{
2082         \bool_if:NTF \amc_num_expovetical_bool {
2083           \hbox{\vbox{
2084             \vbox{\amc_num_integer_boxes:NnnNN
2085               \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
2086               \amc_num_sign_int}
2087             \int_compare:nNnTF \amc_num_expo_int > 0 {
2088               \vspace{\AMCnumeric@Vspace}
2089               \vbox{\hbox{\AMCexponent}}
2090               \vspace{\AMCnumeric@Vspace}
2091               \vbox{\amc_num_integer_boxes:NnnNN
2092                 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
2093                 \amc_num_expo_sign_int}
2094             } {}
2095           }}
2096       } {

```

```

2097 \amc_num_integer_boxes:NnnNN
2098 \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
2099 \amc_num_sign_int
2100 \int_compare:nNnTF \amc_num_expo_int > 0 {
2101 \hspace{\AMCnumeric@Hspace}\AMCexponent\hspace{\AMCnumeric@Hspace}
2102 \amc_num_integer_boxes:NnnNN
2103 \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
2104 \amc_num_expo_sign_int
2105 } {}
2106 }
2107 }
2108 }
2109 \fi

```

And tell AMC that we finished with this question:

```

2110 \ifAMC@ensemble\else\vspace{\AMCpostNquest}\par\fi
2111 \ifAMC@ensemble\ifAMCformulaire@dedans
2112 \AMCmessage{FQ}
2113 \fi\fi
2114 }
2115 }
2116
2117 \cs_new_eq:NN \AMCnumericShow \amc_numeric_show:nn
2118

```

`\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.

```

2119 \cs_new:Npn \amc_numeric_hide:nn #1 #2 {
2120 \keys_set:nn { amcnumeric } { #2 }
2121 \AMCnxtxtGoto
2122 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi
2123 }
2124
2125 \cs_new_eq:NN \AMCnumericHide \amc_numeric_hide:nn
2126
2127 \ExplSyntaxOff
2128 \def\AMCnumericChoicesPlain{%
2129 \AMC@if@separate@question{\AMC@mem@category{numeric}}}%
2130 \AMCformatChoices{\AMCnumericShow}{\AMCnumericHide}%
2131 }

```

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

```

2132 \ExplSyntaxOn
2133
2134 \cs_new:Npn \amc_numeric_choices:nn #1#2 {
2135 \AMCnumericChoicesPlain{#1}{#2}
2136 }
2137 \cs_generate_variant:Nn \amc_numeric_choices:nn { xn }
2138 \cs_new_eq:NN \AMCnumericChoices \amc_numeric_choices:xn
2139
2140 \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 $1/4$?

☐ $[0, 0.1[$
☒ $[0.2, 0.3[$
☐ $[0.4, 0.5[$
☐ $[0.6, 0.7[$
☐ $[0.8, 0.9[$
☐ $[0.1, 0.2[$
☐ $[0.3, 0.4[$
☐ $[0.5, 0.6[$
☐ $[0.7, 0.8[$
☐ $[0.9, 1[$

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

```
2141 \def\AMCIntervalFormat#1#2{[#1,\,#2]}
```

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

```
2142 \ExplSyntaxOn
```

```
2143
```

```
2144 \fp_new:N \amc_interv_a
```

```
2145 \fp_new:N \amc_interv_b
```

```
2146 \cs_new:Npn \amc_intervals:nnnn #1 #2 #3 #4 {
```

```
2147   \fp_set:Nn \amc_interv_a { #2 }
```

```
2148   \fp_do_while:nn { \amc_interv_a < #3 } {
```

```
2149     \fp_set:Nn \amc_interv_b { \amc_interv_a + #4 }
```

```
2150     \fp_compare:nTF { \amc_interv_a <= #1 < \amc_interv_b }
```

```
2151       \correctchoice \wrongchoice
```

```
2152       {\AMCIntervalFormat{\fp_use:N \amc_interv_a}{\fp_use:N \amc_interv_b}}
```

```
2153     \fp_set:Nn \amc_interv_a \amc_interv_b
```

```
2154   }
```

```
2155 }
```

```
2156 \cs_new_eq:NN \AMCIntervals \amc_intervals:nnnn
```

```
2157
```

```
2158 \ExplSyntaxOff
```

4.14 Open questions

`\AMCOpen` The command `\AMCOpen{\langle options \rangle}{\langle choices \rangle}` 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:

```

2159 \def\AMCotextGoto{}
2160 \def\AMCotextReserved{}
2161 \def\AMCocol@Background{lightgray}
2162 \def\AMCocol@BoxFrameRule{white}
2163 \def\AMCocol@FrameRule{black}
2164 \def\AMCocol@Foreground{}
2165 \def\AMCopen@answer{}
2166 \def\AMCopen@question{}
2167 \def\AMCopen@lineuptext{}
2168 \define@key{AMCOpen}{backgroundcol}{\def\AMCocol@Background{#1}}
2169 \define@key{AMCOpen}{foregroundcol}{\def\AMCocol@Foreground{#1}}
2170 \define@key{AMCOpen}{Treserved}{\def\AMCotextReserved{#1}}
2171 \define@key{AMCOpen}{question}{[\AMCid@name]{\def\AMCopen@question{#1}}
2172 \define@key{AMCOpen}{answer}{\def\AMCopen@answer{#1}}
2173 \define@key{AMCOpen}{contentcommand}{[AMCopen@lines]{\def\AMCopen@contentcommand{#1}}
2174 \newdimen\AMCopen@Hspace\AMCopen@Hspace=.5em
2175 \define@key{AMCOpen}{hspace}{\AMCopen@Hspace=#1}
2176 \def\AMCopen@Width{.95\linewidth}
2177 \define@key{AMCOpen}{width}{\def\AMCopen@Width{#1}}
2178 \newdimen\AMCopen@LineHeight\AMCopen@LineHeight=1cm
2179 \define@key{AMCOpen}{lineheight}{\AMCopen@LineHeight=#1}
2180 \newcount\AMCopen@Lines\AMCopen@Lines=1
2181 \define@key{AMCOpen}{lines}{\AMCopen@Lines=#1}
2182 \newdimen\AMCopen@boxmargin\AMCopen@boxmargin=3pt
2183 \define@key{AMCOpen}{boxmargin}{\AMCopen@boxmargin=#1}
2184 \newdimen\AMCopen@boxframerule\AMCopen@boxframerule=1pt
2185 \define@key{AMCOpen}{boxframerule}{\AMCopen@boxframerule=#1}
2186 \define@key{AMCOpen}{boxframerulecol}{\def\AMCocol@BoxFrameRule{#1}}
2187 \define@key{AMCOpen}{framerulecol}{\def\AMCocol@FrameRule{#1}}
2188 \newdimen\AMCopen@framerule\AMCopen@framerule=1pt
2189 \define@key{AMCOpen}{framerule}{\AMCopen@framerule=#1}
2190 \define@key{AMCOpen}{lineuptext}{\def\AMCopen@lineuptext{#1}}
2191 \define@boolkey{AMCOpen}{dots}[true]{}
2192 \define@boolkey{AMCOpen}{scan}[true]{}
2193 \define@boolkey{AMCOpen}{retick}[true]{}

```

```

2194 \define@boolkey{AMCOpen}{annotate}[false]{}
2195 \define@boolkey{AMCOpen}{lineup}[false]{}
2196 \setkeys{AMCOpen}{dots,scan,retick,annotate,lineup,contentcommand}
2197 \newcommand\AMCopenOpts[1]{\setkeys{AMCOpen}{#1}}

```

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

```

2198 \newcommand\AMCopen@lines{%
2199   \begin{minipage}{\AMCopen@Width}%
2200     \loop\vspace{\AMCopen@LineHeight}
2201     \hspace*{.5em}\ifAMC@correc\smash{\AMCopen@answer}\def\AMCopen@answer{}\fi%
2202     \ifKV@AMCopen@dots%
2203       \dotfill\hspace*{.5em}
2204     \fi
2205     \ifnum\AMCopen@Lines>\@ne\par\advance\AMCopen@Lines\m@ne\repeat%
2206   \end{minipage}
2207 }
2208 \newcommand\AMCopenShow[2]{
2209   \ifAMC@ensemble\ifAMC@formulaire@dedans%
2210     \AMCmessage{Q=\the\AMCid@quest}%
2211     \fi\fi%
2212   {\setkeys{AMCOpen}{#1}%
2213     \ifKV@AMCopen@lineup%
2214       \ifAMC@ensemble\else%
2215         \ifx\@empty\AMCopen@lineuptext\@empty\fi%
2216       \fi%
2217       \ifAMC@correc\smash{\AMCopen@answer}\fi%
2218       \ifx\@empty\AMCopen@lineuptext\@empty%
2219         \dotfill%
2220       \else%
2221         \AMCopen@lineuptext\hfill%
2222       \fi%
2223     \else%
2224       \hspace*{.5em}\linebreak[1]\hspace*{\fill}%
2225     \fi%
2226     {\AMCnoCompleteMulti%
2227       \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2228       \def\AMCanswer##1##2{\ifAMC@ensemble ##1\else%
2229         \ifAMC@inside@box ##1\else\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2230       \hspace{\AMCopen@Hspace}}%
2231       \fboxsep=\AMCopen@boxmargin%
2232       \fboxrule=\AMCopen@boxframerule%
2233       \fcolorbox{\AMCocol@BoxFrameRule}{\AMCocol@Background}{%
2234         \ifAMC@ensemble\AMCopen@question%
2235           \ifx\@empty\AMCopen@question\@empty\else\hspace{\AMCopen@Hspace}\fi%
2236         \fi%
2237         \begin{choicescustom}[o]%
2238           \ifx\AMCocol@Foreground\@empty\@empty\else%
2239             \def\AMC@boxcolor{\AMCocol@Foreground}%
2240           \fi%
2241           #2%
2242           \ifKV@AMCopen@scan\else\AMCdontScan\fi%
2243           \ifKV@AMCopen@retick\AMCreTick\fi%
2244           \ifKV@AMCopen@annotate\else\AMCdontAnnotate\fi%

```



```

2245 \end{choicescustom}%
2246 \ifx\@empty\AMCotextReserved\@empty%
2247 \hspace{-\AMCopen@Hspace}%
2248 \else%
2249 \ifx\AMCocol@Foreground\@empty\@empty%
2250 \AMCotextReserved%
2251 \else%
2252 \textcolor{\AMCocol@Foreground}{\AMCotextReserved}%
2253 \fi%
2254 \fi%
2255 }}%
2256 \ifKV@AMCOpen@lineup\else%
2257 \par\nobreak\noindent%
2258 \hspace*{\fill}{%
2259 \fboxrule=\AMCopen@framerule%
2260 \fcolorbox{\AMCocol@FrameRule}{white}{%
2261 \csname\AMCopen@contentcommand\endcsname
2262 }}%
2263 \vspace{\AMCpost@quest}\par%
2264 \fi%
2265 }%
2266 \ifAMC@ensemble\ifAMC@formulaire@dedans%
2267 \AMCmessage{FQ}%
2268 \fi\fi%
2269 }
2270 \newcommand\AMCopenHide[2]{%
2271 \AMCotextGoto%
2272 \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2273 }
2274 \def\AMCOpen{%
2275 \AMC@if@separate@question{\AMC@mem@category{open}}%
2276 \AMCformatChoices{\AMCopenShow}{\AMCopenHide}%
2277 }

```

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}

2278 \def\AMCbotextGoto{}
2279 \def\AMCbo@help{}
2280 \define@key{AMCBoxOnly}{help}{\def\AMCbo@help{#1}}
2281 \define@boolkey{AMCBoxOnly}{ordered}[false]{}
2282 \setkeys{AMCBoxOnly}{ordered}
2283 \newcommand\AMCboOpts[1]{\setkeys{AMCBoxOnly}{#1}}

```

```

2284 \newcommand\AMCboShow[2]{%
2285   \ifAMC@ensemble\ifAMCformulaire@dedans%
2286     \AMCmessage{Q=\the\AMCid@quest}%
2287   \fi\fi%
2288   {\setkeys{AMCBoxOnly}{#1}%
2289     \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2290     \def\AMCanswer##1##2{\hspace{\AMCformHSpace} \ifAMC@ensemble ##1\else%
2291       \ifAMC@inside@box ##1\else{\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2292     }%
2293     \ifAMC@ensemble\AMCbo@help\fi%
2294     \ifKV@AMCBoxOnly@ordered%
2295       \begin{choicescustom}[o]%
2296     \else%
2297       \begin{choicescustom}%
2298     \fi%
2299     #2
2300   \end{choicescustom}%
2301 }%
2302 \ifAMC@ensemble\ifAMCformulaire@dedans%
2303   \AMCmessage{FQ}%
2304   \fi\fi%
2305 }
2306 \newcommand\AMCboHide[2]{
2307   \AMCbotextGoto%
2308   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2309 }
2310 \def\AMCBoxOnly{%
2311   \AMC@if@separate@question{\AMC@mem@category{box}}%
2312   \AMCformatChoices{\AMCboShow}{\AMCboHide}%
2313 }

```

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.

```

2314 \DeclareFontShape{OT1}{cmr}{b}{n}{<35->cmr17}{-}
2315 \def\AMC@watertext{\AMC@loc@draft}
2316 \newcommand\AMCw@termark{%
2317   \setlength{\@tempdimb}{.5\paperwidth}%
2318   \setlength{\@tempdimc}{-.5\paperheight}%
2319   \put(\strip@pt\@tempdimb,\strip@pt\@tempdimc){%
2320     \makebox(0,0){\rotatebox{45}{\AMC@LR{%
2321       \textcolor{gray}{0.8}{
2322         \fontencoding{OT1}\fontfamily{cmr}
2323         \fontseries{b}\fontshape{n}
2324         \fontsize{90pt}{120pt}
2325         \selectfont
2326         \AMC@watertext}}}}}}
2327 \newcommand\AMCw@terprint[1]{%
2328   \setbox\@tempboxa\vbox to \z@{%
2329     \vbox{%

```

```

2330 \hbox to \z@{%
2331   #1\hss}\vss}
2332 \dp\@tempboxa\z@
2333 \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}.

```

2334 \def\AMCcircle#1#2{%
2335   {\setlength{\unitlength}{1mm}%
2336     \begin{picture} (#1,#1) (-#2,-#2) \thinlines \circle*{#1} \end{picture}}
2337 \def\m@rqueCalage{\AMCcircle{3.6}{1.8}}
2338 \def\m@rque#1{\AMC@tracebox{1}{#1}{\m@rqueCalage}}
2339 \def\he@dtaille#1{% \par cancels the \leavevmode
2340   % introduced by https://github.com/pietvo/fancyhdr/commit/6b1ad10eeb5bc3d804f3cd2cf193e6440d0229e6
2341   \par\vbox to 1cm{#1}}
2342 \def\he@dbas#1{\he@dtaille{\vspace*{\fill}#1}}
2343 \def\he@dhaut#1{\he@dtaille{#1\vspace*{\fill}}}
2344 \def\AMC@intituleHead{\AMC@loc@corrected}
2345 \def\AMC@note{}
2346 \def\AMCsetFoot#1{\def\AMC@note{#1}}
2347 \newcommand\AMCStudentNumber{\the\AMCid@etud}
2348 \newcommand\AMCIDBoxesA{\AMC@binaryCode{id=1,ndigits=\AMC@NCBetud}{\the\AMCid@etud}}
2349 \newcommand\AMCIDBoxesB{\AMC@binaryCode{id=2,ndigits=\AMC@NCBpage}{\thepage}}
2350 \newcommand\AMCIDBoxesC{\AMC@binaryCode{id=3,ndigits=\AMC@NCBcheck}{\the\AMCid@check}}
2351 \newcommand\AMCIDBoxesABC{%
2352   \hbox{\vbox{\noindent\AMCIDBoxesA\
2353     \noindent\AMCIDBoxesB\AMCIDBoxesC}}}%
2354 }
2355 \def\AMC@pageHook{%
2356   \AMC@pagepos%
2357   \ifAMC@pagelayout\global\advance\AMCid@check\m@ne%
2358   \ifnum\AMCid@check<1\global\AMCid@check=\AMCid@checkmax\fi%
2359   \ifAMC@watermark\ifAMC@correthead\else\AMCwaterprint{\AMCwatermark}%
2360   \fi\fi\fi}
2361 \@ifl@t@r{\fmtversion{2020/10/01}}
2362   {\AddToHook{shipout/background}{\put(0in,0in){\AMC@pageHook}}}
2363   {\AtBeginShipout{\AtBeginShipoutUpperLeft{\AMC@pageHook}}}
2364 \fancypagestyle{AMCpageHeadOnly}{%
2365   \fancyhf{} \fancyhead[C]{\textsc{\AMC@intituleHead}}%
2366   \renewcommand{\headrulewidth}{0pt}%
2367   \renewcommand{\footrulewidth}{0pt}%
2368 }
2369 \fancypagestyle{AMCpageFull}{%
2370   \fancyhf{}%
2371   \fancyhead[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHG}}}}%

```

```

2372 \fancyhead[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHD}}}}%
2373 \fancyfoot[L]{\AMC@LR{\leavevmode\m@rque{positionBG}}}%
2374 \fancyfoot[R]{\AMC@LR{\leavevmode\m@rque{positionBD}}}%
2375 \fancyhead[C]{\AMC@LR{\he@dhaut{%
2376     \begin{minipage}[b]{\AMC@CBtaille}\AMCboxColor{black}%
2377     \ifAMCids@top\vbox to \AMCids@height{\texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2378     \AMCIDBoxesABC
2379     \end{minipage}%
2380     \ifAMCids@side\hbox to \AMCids@width{\hspace*{\fill}%
2381     \texttt{+\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2382     }}}%
2383 \fancyhfoffset[EOLR]{5mm}%
2384 \fancyfoot[C]{\AMC@note}%
2385 \renewcommand{\headrulewidth}{0pt}%
2386 \renewcommand{\footrulewidth}{0pt}%
2387 }
2388 \newcommand\AMCsubjectPageTag{%
2389     \fbox{\texttt{\the\AMCid@etud:\thepage}}%
2390 }
2391 \fancypagestyle{AMCpageNoMarks}{%
2392     \fancyhf{}%
2393     \fancyhead[R]{\AMCsubjectPageTag}%
2394     \fancyfoot[C]{\AMC@note}%
2395     \renewcommand{\headrulewidth}{0pt}%
2396     \renewcommand{\footrulewidth}{0pt}%
2397 }
2398 \fancypagestyle{AMCpageEmpty}{%
2399     \fancyhf{}%
2400     \renewcommand{\headrulewidth}{0pt}%
2401     \renewcommand{\footrulewidth}{0pt}%
2402 }
2403 \AtBeginDocument{%
2404     \ifAMC@pagelayout%
2405         \ifAMC@correthead
2406             \pagestyle{AMCpageHeadOnly}
2407         \else
2408             \pagestyle{AMCpageFull}
2409         \fi
2410     \fi
2411 }

```

4.17 Defining a single exam copy content

`\onecopy` The command `\onecopy[$\langle n \rangle$]{ $\langle code \rangle$ }` generates $\langle n \rangle$ copies of the subject that is described in $\langle code \rangle$. The L^AT_EX code $\langle code \rangle$ that generates a single copy can be a little long, so that the environment `examcopy` is often preferred.

```

2412 \newcommand{\onecopy}[2]{%
2413     \ifx\AMCNombreCopies\undefined\AMCnum@copies=#1%
2414     \else\AMCnum@copies=\AMCNombreCopies\fi%
2415     \AMCmessage{TOTAL=\the\AMCnum@copies}%
2416     \message{^^JAMC:copies:total=\the\AMCnum@copies^^J}%
2417     \AMCid@etud=\AMCid@etudstart%
2418     \ifnum\AMCid@etud=0\AMCid@etud=\AMC@premierecopie\fi%

```

```

2419 \AMCid@etudfin=\AMCnum@copies%
2420 \advance\AMCid@etudfin\AMCid@etud\relax%
2421 \ifAMC@correthead\AMCid@etudfin=\AMC@premierecopie%
2422 \message{^^JAMC:copies:total=1^^J}%
2423 \fi
2424 \ifAMC@pdfform\begin{Form}\fi%
2425 \loop{%
2426 \global\AMCrep@nnmax=0%
2427 \ifAMC@calibration\protected@write\AMC@XYFILE{}\fi%
2428 \string\rngstate{\the\AMCid@etud}\the\AMC@SR}%
2429 }\fi%
2430 \AMC@zoneformulairefalse\setcounter{page}{1}\setcounter{section}{0}%
2431 \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageNoMarks}\fi\fi%
2432 \AMCnumero{1}%
2433 \ifAMC@calibration\AMCmessage{ETU=\the\AMCid@etud}\fi%
2434 \AMC@multiclear%
2435 \global\AMC@keepmemoryfalse%
2436 #2%
2437 \ifAMC@keepmemory\else\AMC@mem@clear\fi%
2438 \clearpage}%
2439 \message{^^JAMC:copies:add=1^^J}%
2440 \advance\AMCid@etud\@one\ifnum\AMCid@etud<\AMCid@etudfin\repeat%
2441 \global\AMCid@etudstart=\AMCid@etud%
2442 \ifAMC@pdfform\end{Form}\fi%
2443 \AMC@multi@report%
2444 }

```

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

```

2445 \newcount\AMC@addpages
2446 \newcommand{\AMCaddpagesto}[1]{%
2447 \AMC@addpages=#1\advance\AMC@addpages\@one%
2448 \clearpage%
2449 \@whilenum\thepage<\AMC@addpages\do{%
2450 \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2451 \hbox{}}\clearpage%
2452 }%
2453 }

```

\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).

```

2454 \def\AMCcleardoublepage{%
2455 \clearpage%
2456 \ifodd\thepage\else%
2457 \ifAMC@automarks\pagestyle{AMCpageEmpty}\fi%
2458 \hbox{}}\clearpage%
2459 \fi%
2460 }

```

\exemplairepair To make some differences in the copies, checking if the student sheet number is odd, with

`\exemplairepair` construct, can be useful.

```
2461 \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
2462 \newcommand\AMCstudentlabel[1]{\the\AMCid@etud-#1}
2463 \def\AMClabel#1{\expandafter\label{\AMCstudentlabel{#1}}}
2464 \def\AMCref#1{\expandafter\ref{\AMCstudentlabel{#1}}}
2465 \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.

```
2466 \newcommand{\AMCqlabel}[1]{%
2467   \AMClabel{#1}%
2468 }
```

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.

```
2469 \newcommand{\AMCassociation}[2][]{%
2470   \ifAMC@calibration%
2471     \immediate\write\AMC@XYFILE{\string\association{\the\AMCid@etud}{#2}{#1}}%
2472   \fi%
2473 }
```

`\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>}`.

```
2474 \newcommand{\AMCstudentslistfile}[2]{%
2475   \ifAMC@calibration%
2476     \immediate\write\AMC@XYFILE{\string\with{studentslistfile=#1}}%
2477     \immediate\write\AMC@XYFILE{\string\with{studentslistkey=#2}}%
2478   \fi%
2479 }
```

4.19 Package options

See section 3.1 for the options descriptions.

```
2480 \def\AMC@lang@code{}
2481 \DeclareOptionX{noshuffle}{\AMC@ordretrue}
2482 \DeclareOptionX{noshufflegroups}{\AMC@shuffleGfalse}
2483 \DeclareOptionX{fullgroups}{\AMC@fullGroupstrue}
2484 \DeclareOptionX{answers}{\AMC@corretheadtrue\AMC@correcttrue}
2485 \DeclareOptionX{indivanswers}{\AMC@correcttrue}
2486 \DeclareOptionX{textpos}{\AMC@textPostrue}
2487 \DeclareOptionX{extractonly}{\AMC@extractOnlytrue\AMC@textPostrue\AMC@boxStyle{shape=none}\AMC@boxedAnswer}
2488 \DeclareOptionX{box}{\AMC@qbloctrue}
2489 \DeclareOptionX{asbox}{\AMC@asqbloctrue}
2490 \DeclareOptionX{separateanswersheet}{\AMC@ensembletrue}
2491 \DeclareOptionX{digits}{\AMC@inside@digittrue}
```

```

2492 \DeclareOptionX{ordre}{\AMC@ordretrue}
2493 \DeclareOptionX{correc}{\AMC@corretheadtrue\AMC@correcttrue}
2494 \DeclareOptionX{modele}{\AMC@corretheadtrue\AMC@correcfalse\AMC@ordretrue}
2495 \DeclareOptionX{correcindiv}{\AMC@correcttrue}
2496 \DeclareOptionX{init}{\AMC@SR@time}
2497 \DeclareOptionX{bloc}{\AMC@qbloctrue}
2498 \DeclareOptionX{completemulti}{\AMC@complete@multitrue}
2499 \DeclareOptionX{insidebox}{\AMC@inside@boxtrue}
2500 \DeclareOptionX{ensemble}{\AMC@ensembletrue}
2501 \DeclareOptionX{chiffres}{\AMC@inside@digittrue}
2502 \DeclareOptionX{outsidebox}{\AMC@outside@boxtrue}
2503 \DeclareOptionX{calibration}{\AMC@calibrationtrue}
2504 \DeclareOptionX{nowatermark}{\AMC@watermarkfalse}
2505 \newcommand\AMC@catalogMode{%
2506   \AMC@catalogtrue%
2507   \AMC@watermarkfalse\AMC@corretheadtrue%
2508   \AMC@correcttrue\AMC@ordretrue\AMC@shuffleGfalse%
2509   \AMC@fullGroupstrue%
2510   \def\AMC@intituleHead{\AMC@loc@catalog}\AMC@affichekeystrue}
2511 \newcommand\AMC@keys@next{\AMC@keyslinefalse}
2512 \newcommand\AMC@keys@line{\AMC@keyslinetrue}
2513 \DeclareOptionX{catalog}{\AMC@catalogMode}
2514 \DeclareOptionX{keys}[next]{\csname AMC@keys@#1\endcsname{}}
2515 \DeclareOptionX{francais}{\def\AMC@lang@code{FR}\AMC@loc@FR}
2516 \DeclareOptionX{lang}{\def\AMC@lang@code{#1}\csname AMC@loc@#1\endcsname}
2517 \DeclareOptionX{versionA}{%
2518   \def\AMC@id@checkmax{31}\def\AMC@NCBetud{9}\def\AMC@NCBpage{4}%
2519   \def\AMC@NCBcheck{5}\setlength{\AMC@CBtaille}{4cm}%
2520   \def\AMC@premierecopie{100}}
2521 \DeclareOptionX{plain}{\AMC@plaintrue}
2522 \DeclareOptionX{nopage}{\AMC@pagelayoutfalse}
2523 \DeclareOptionX{postcorrect}{\AMC@postcorrecttrue}
2524 \DeclareOptionX{automarks}{\AMC@automarkstrue}
2525 \newif\ifAMCneeds@storebox\AMCneeds@storeboxfalse
2526 \DeclareOptionX{storebox}{\AMCneeds@storeboxtrue}
2527 \DeclareOptionX{pdfform}{\AMC@pdfformtrue}
2528 \DeclareOptionX{codedigit}{\AMC@codeID@{#1}}
2529 \newif\ifAMC@survey\AMC@surveyfalse
2530 \DeclareOptionX{survey}{\AMC@surveytrue}
2531 \ProcessOptionsX
2532 \ifAMCneeds@storebox
2533   \RequirePackage{storebox}\AtBeginDocument{{}}%
2534 \fi
2535 \ifAMC@pdfform
2536   \AMCmessage{VAR:project:pdfform=1}%
2537   \AMCboxStyle{shape=form}%
2538   \RequirePackage[pageanchor=false]{hyperref}%
2539 \else%
2540   \AMCmessage{VAR:project:pdfform=0}%
2541 \fi
2542 \AtBeginDocument{%
2543   \ifAMCneeds@storebox%
2544     \let\AMC@new@savebox=\newstorebox%

```

```

2545 \let\AMC@save@box=\storebox%
2546 \let\AMC@use@box=\usestorebox%
2547 \fi%
2548 \AMC@new@savebox{\AMC@ovalbox@R}%
2549 \AMC@new@savebox{\AMC@ovalbox@RF}%
2550 \AMC@new@savebox{\AMC@ovalbox@}%
2551 \AMC@new@savebox{\AMC@ovalbox@F}%
2552 \AMC@shapeprepare%
2553 }

```

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.) 2554 \ifAMC@survey
question-auto (env.) 2555 \NewEnviron{Questionnaires}[1]{
  values (env.) 2556 \onecopy{#1}{
  values-auto (env.) 2557 \BODY
variable-auto (env.) 2558 }
  \answer 2559 }
  2560 \NewEnviron{auto}[1]{
  2561 \csvreader[head to column names, separator=tab]{#1}{}{
  2562 \BODY
  2563 }
  2564 }
  2565 \NewEnviron{question-auto}[3]{
  2566 \csvreader[head to column names, separator=tab]{#1}{}{
  2567 \ifcsvstrcmp{#2}{#3}{\BODY \\\}
  2568 }
  2569 }
  2570 \newenvironment{values}{}{}
  2571 \NewEnviron{values-auto}[5]{
  2572 \csvreader[head to column names, separator=tab]{#1}{}{
  2573 \ifcsvstrcmp{#2}{#3}{
  2574 \ifcsvstrcmp{#4}{#5}{\BODY \\\}
  2575 }{}
  2576 }{}
  2577 }
  2578 }
  2579 \NewEnviron{variable-auto}[3]{
  2580 \foreach \x in {#3}{
  2581 \csvreader[head to column names, separator=tab]{#1}{}{
  2582 \ifcsvstrcmp{#2}{\x}{\BODY}{
  2583 }
  2584 }
  2585 }
  2586 \newcommand{\answer}[5][\global\advance\AMCrep@count\@ne\relax%
  2587 \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%
  2588 \global\AMCune@bonnettrue%
  2589 \AMCload@response{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkbox}%

```



```

2590 \else\AMC@box{#1}{\fi}{#2}{#3}{#4}{#5}}{\the\AMCrep@count}\ignorespaces}
2591 \RequirePackage{tikz}
2592 \usetikzlibrary{positioning, shapes, arrows, tikzmark, decorations.pathreplacing}
2593 \tikzset{
2594     checkbox-sc/.style={
2595         right=of lab\thecsvrow
2596     },
2597     vallab-sc/.style={
2598         text width=4cm,
2599         align=left,
2600     },
2601     checkbox-mc/.style={
2602     },
2603     vallab-mc/.style={
2604         above=of box\thecsvrow,
2605         text width=1.4cm,
2606         align=center,
2607     },
2608     varlab-mc/.style={
2609         text width=4cm,
2610         align=left,
2611     },
2612     node distance= 0mm,
2613 }
2614 \fi

```

4.21 Package Errors

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

```

2615 \def\AMC@error@explain{\PackageError{automultiplechoice}{
2616     Command \protect\explain\space can only be used inside\MessageBreak question like environments}{Some
2617 }}

```

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.

```

2618 \ifAMC@plain
2619 \else
2620     \IfFileExists{environ.sty}{\RequirePackage{environ}}{}
2621     \ifx\TeXversion\@undefined
2622     \else
2623         \RequirePackage{etex}
2624     \fi
2625 \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`.

```

2626 \@ifpackageloaded{environ}{%

```

```

2627 \ifx\NewEnviron\undefined\PackageWarning{automultiplechoice}%
2628 {Package environ loaded but too old version:
2629     environnement examcopy/copieexamen will NOT be defined.}%
2630 \else\NewEnviron{examcopy}[1][5]{\onecopy{#1}{\BODY}}\fi}%
2631 {\PackageWarning{automultiplechoice}%
2632 {Package environ not loaded: environnement
2633     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.

```

2634 \ExplSyntaxOn
2635
2636 \cs_if_exist:NTF \pdfsavepos { } {
2637   \cs_if_exist:NTF \savepos { \cs_new_eq:NN \pdfsavepos \savepos } { }
2638 }
2639 \cs_if_exist:NTF \pdflastxpos { } {
2640   \cs_if_exist:NTF \lastxpos { \cs_new_eq:NN \pdflastxpos \lastxpos } { }
2641 }
2642 \cs_if_exist:NTF \pdflastypos { } {
2643   \cs_if_exist:NTF \lastypos { \cs_new_eq:NN \pdflastypos \lastypos } { }
2644 }

```

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

```

2645
2646 \cs_if_exist:NTF \pdfpagewidth { } {
2647   \cs_new_eq:NN \pdfpagewidth \paperwidth
2648 }
2649 \cs_if_exist:NTF \pdfpageheight { } {
2650   \cs_new_eq:NN \pdfpageheight \paperheight
2651 }
2652
2653 \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 L^AT_EX file: `\CorrigeExterne`
`corrigeIndivExterne` `pdflatex '\nonstopmode\def\SujetExterne{1}\def\NoWatermarkExterne{1}\input{mcq.tex}'`
`NoWatermarkExterne`

```

2654 \ifx\SujetExterne\undefined\else
2655 \message{***SUJET***^J}
2656 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse
2657 \fi
2658 \ifx\ScoringExterne\undefined\else
2659 \message{***SCORING***^J}
2660 \AMC@calibrationtrue\AMC@correcfalse\AMC@corretheadfalse\AMC@watermarkfalse\AMC@invisibletrue
2661 \fi

```

```

2662 \ifx\CorrigeExterne\undefined\else
2663 \message{***CORRIGE***^^J}
2664 \AMC@calibrationfalse\AMC@corretheadtrue\AMC@correcttrue\AMC@watermarkfalse
2665 \fi
2666 \ifx\CorrigeIndivExterne\undefined\else
2667 \message{***CORRIGE***^^J}
2668 \AMC@calibrationtrue\AMC@corretheadfalse\AMC@correcttrue\AMC@watermarkfalse
2669 \fi
2670 \ifx\CatalogExterne\undefined\else
2671 \message{***CATALOG***^^J}
2672 \AMC@catalogMode
2673 \fi
2674 \ifx\NoWatermarkExterne\undefined\else
2675 \AMC@watermarkfalse
2676 \fi
2677 \ifx\codeDigitExterne\undefined\else
2678 \AMCcodeID@{\codeDigitExterne}
2679 \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:

```

2680 \@ifpackageloaded{geometry}{\usepackage{geometry}}
2681 \ifAMC@pagelayout
2682   \ifAMC@correthead
2683     \geometry{hmargin=3cm,vmargin={1cm,1cm},includeheadfoot,headheight=1cm,footskip=1cm}
2684   \else
2685     \geometry{hmargin=3cm,headheight=2cm,headsep=.3cm,footskip=1cm,top=3.5cm,bottom=2.5cm}
2686   \fi
2687   \ifAMC@watermark
2688     \ifAMC@correthead\else
2689       \def\AMC@note{\begin{minipage}{0.65\linewidth}
2690         \AMC@LR{\textcolor{blue}{\AMC@loc@message}}
2691       \end{minipage}}
2692     }
2693   \fi
2694 \fi
2695 \fi

```

4.26 Initialisation

Initialisation of the check counter:

```
2696 \AMCid@check=\AMCid@checkmax
```

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

```

2697 \ifAMC@ensemble\AMCmessage{VAR:ensemble=1}\fi
2698 \ifAMC@inside@box\AMCmessage{VAR:insidebox=1}\fi
2699 \ifAMC@outside@box\AMCmessage{VAR:outsidebox=1}\fi
2700 \ifAMC@postcorrect\AMCmessage{VAR:postcorrect=1}\fi

```

Preparing writing to .xy file :

```
2701 \ifAMC@calibration
```

```

2702 \newwrite\AMC@XYFILE%
2703 \immediate\openout\AMC@XYFILE\jobname.xy%
2704 \immediate\write\AMC@XYFILE{\string\version{\AMC@VERSION}}
2705 \immediate\write\AMC@XYFILE{\string\with{codedigit=\AMCcodeID@mode}}
2706 \immediate\write\AMC@XYFILE{\string\with{version=\AMC@VERSION}}
2707 \immediate\write\AMC@XYFILE{\string\with{ensemble=\ifAMC@ensemble yes\else no\fi}}
2708 \immediate\write\AMC@XYFILE{\string\with{insidebox=\ifAMC@inside@box yes\else no\fi}}
2709 \immediate\write\AMC@XYFILE{\string\with{outsidebox=\ifAMC@outside@box yes\else no\fi}}
2710 \immediate\write\AMC@XYFILE{\string\with{postcorrect=\ifAMC@postcorrect yes\else no\fi}}
2711 \immediate\write\AMC@XYFILE{\string\with{extractonly=\ifAMC@extractOnly yes\else no\fi}}
2712 \immediate\write\AMC@XYFILE{\string\with{lang=\AMC@lang@code}}
2713 \ifx\AMCNombreCopies\undefined%
2714 \immediate\write\AMC@XYFILE{\string\with{ncopies=default}}}%
2715 \else%
2716 \immediate\write\AMC@XYFILE{\string\with{ncopies=\AMCNombreCopies}}}%
2717 \fi%
2718 \fi

```

4.27 French command names

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

```

2719 \let\reponses=\choices\let\endreponses=\endchoices
2720 \let\reponseshoriz=\choiceshoriz\let\endreponseshoriz=\endchoiceshoriz
2721 \let\reponsesperso=\choicescustom\let\endreponsesperso=\endchoicescustom
2722 \let\bonne=\correctchoice
2723 \let\mauvaise=\wrongchoice
2724 \let\bareme=\scoring
2725 \let\baremeDefaultM=\scoringDefaultM
2726 \let\baremeDefaultS=\scoringDefaultS
2727 \def\exemplaire{\AMC@loc@FR\onecopy}
2728 \@ifpackageloaded{environ}{%
2729   \let\copieexamen=\examcopy\let\endcopieexamen=\endexamcopy}{%
2730 \let\melangegroupe=\shufflegroup
2731 \let\restituegroupe=\insertgroup
2732 \let\alafin=\lastchoices
2733 \let\formulaire=\AMCform
2734 \let\AMCdebutFormulaire=\AMCformBegin
2735 \let\champnom=\namefield
2736 \let\choixIntervalles=\AMCIntervals

```

5 Outputs

In the .xy file, $0/\langle n \rangle$ 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 $\langle n \rangle$ inside this student sheet. Then, each instance of the `\tracepos` command shows x and y positions as arguments #2 and #3 (unit is `sp`, such that $65536 \times 72.27 \text{ sp}$ is one inch). One has to take min and max of the x -values to determine the left and right position of the box, and min and max values of y -values to determine top and bottom position of the box.

5.1 namefield command

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

```

\tracepos{0/35: __zone:id: __n}{0sp}{39236125sp}{square}
\tracepos{0/35: __zone:id: __n}{5873801sp}{0sp}{square}
\tracepos{0/35: __zone:id: __n}{15861297sp}{0sp}{square}
\tracepos{0/35: __zone:id: __n}{0sp}{36250947sp}{square}

```

5.2 AMCboxedchar command

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

```

\tracepos{0/35:test}{22855914sp}{23116900sp}{square}
\tracepos{0/35:test}{23561334sp}{22411480sp}{square}

```

5.3 AMCcode command

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

```

\tracepos{0/60: case: code [5] : 16, 1}{24875504sp}{27248508sp}{square}
\tracepos{0/60: case: code [5] : 16, 1}{25580924sp}{26543088sp}{square}
\boxchar{0/60: case: code [5] : 16, 1}{A}
\tracepos{0/60: case: code [5] : 16, 2}{24875504sp}{26134396sp}{square}
\tracepos{0/60: case: code [5] : 16, 2}{25580924sp}{25428976sp}{square}
\boxchar{0/60: case: code [5] : 16, 2}{B}
\tracepos{0/60: case: code [5] : 16, 3}{24875504sp}{25020284sp}{square}
\tracepos{0/60: case: code [5] : 16, 3}{25580924sp}{24314864sp}{square}
\boxchar{0/60: case: code [5] : 16, 3}{C}
\tracepos{0/60: case: code [5] : 16, 4}{24875504sp}{23906172sp}{square}
\tracepos{0/60: case: code [5] : 16, 4}{25580924sp}{23200752sp}{square}
\boxchar{0/60: case: code [5] : 16, 4}{D}
\tracepos{0/60: case: code [4] : 17, 1}{26540303sp}{29476732sp}{square}
\tracepos{0/60: case: code [4] : 17, 1}{27245723sp}{28771312sp}{square}
\boxchar{0/60: case: code [4] : 17, 1}{0}
\tracepos{0/60: case: code [4] : 17, 2}{26540303sp}{28362620sp}{square}
\tracepos{0/60: case: code [4] : 17, 2}{27245723sp}{27657200sp}{square}
\boxchar{0/60: case: code [4] : 17, 2}{1}
\tracepos{0/60: case: code [4] : 17, 3}{26540303sp}{27248508sp}{square}
\tracepos{0/60: case: code [4] : 17, 3}{27245723sp}{26543088sp}{square}
\boxchar{0/60: case: code [4] : 17, 3}{2}
\tracepos{0/60: case: code [4] : 17, 4}{26540303sp}{26134396sp}{square}
\tracepos{0/60: case: code [4] : 17, 4}{27245723sp}{25428976sp}{square}
\boxchar{0/60: case: code [4] : 17, 4}{3}
\tracepos{0/60: case: code [4] : 17, 5}{26540303sp}{25020284sp}{square}
\tracepos{0/60: case: code [4] : 17, 5}{27245723sp}{24314864sp}{square}
\boxchar{0/60: case: code [4] : 17, 5}{4}
\tracepos{0/60: case: code [4] : 17, 6}{26540303sp}{23906172sp}{square}
\tracepos{0/60: case: code [4] : 17, 6}{27245723sp}{23200752sp}{square}
\boxchar{0/60: case: code [4] : 17, 6}{5}
\tracepos{0/60: case: code [3] : 18, 1}{28032160sp}{29476732sp}{square}

```

```

\tracepos{0/60:case:code[3]:18,1}{28737580sp}{28771312sp}{square}
\boxchar{0/60:case:code[3]:18,1}{0}
\tracepos{0/60:case:code[3]:18,2}{28032160sp}{28362620sp}{square}
\tracepos{0/60:case:code[3]:18,2}{28737580sp}{27657200sp}{square}
\boxchar{0/60:case:code[3]:18,2}{1}
\tracepos{0/60:case:code[3]:18,3}{28032160sp}{27248508sp}{square}
\tracepos{0/60:case:code[3]:18,3}{28737580sp}{26543088sp}{square}
\boxchar{0/60:case:code[3]:18,3}{2}
\tracepos{0/60:case:code[3]:18,4}{28032160sp}{26134396sp}{square}
\tracepos{0/60:case:code[3]:18,4}{28737580sp}{25428976sp}{square}
\boxchar{0/60:case:code[3]:18,4}{3}
\tracepos{0/60:case:code[3]:18,5}{28032160sp}{25020284sp}{square}
\tracepos{0/60:case:code[3]:18,5}{28737580sp}{24314864sp}{square}
\boxchar{0/60:case:code[3]:18,5}{4}
\tracepos{0/60:case:code[3]:18,6}{28032160sp}{23906172sp}{square}
\tracepos{0/60:case:code[3]:18,6}{28737580sp}{23200752sp}{square}
\boxchar{0/60:case:code[3]:18,6}{5}
\tracepos{0/60:case:code[2]:19,1}{29524017sp}{29476732sp}{square}
\tracepos{0/60:case:code[2]:19,1}{30229437sp}{28771312sp}{square}
\boxchar{0/60:case:code[2]:19,1}{0}
\tracepos{0/60:case:code[2]:19,2}{29524017sp}{28362620sp}{square}
\tracepos{0/60:case:code[2]:19,2}{30229437sp}{27657200sp}{square}
\boxchar{0/60:case:code[2]:19,2}{1}
\tracepos{0/60:case:code[2]:19,3}{29524017sp}{27248508sp}{square}
\tracepos{0/60:case:code[2]:19,3}{30229437sp}{26543088sp}{square}
\boxchar{0/60:case:code[2]:19,3}{2}
\tracepos{0/60:case:code[2]:19,4}{29524017sp}{26134396sp}{square}
\tracepos{0/60:case:code[2]:19,4}{30229437sp}{25428976sp}{square}
\boxchar{0/60:case:code[2]:19,4}{3}
\tracepos{0/60:case:code[2]:19,5}{29524017sp}{25020284sp}{square}
\tracepos{0/60:case:code[2]:19,5}{30229437sp}{24314864sp}{square}
\boxchar{0/60:case:code[2]:19,5}{4}
\tracepos{0/60:case:code[2]:19,6}{29524017sp}{23906172sp}{square}
\tracepos{0/60:case:code[2]:19,6}{30229437sp}{23200752sp}{square}
\boxchar{0/60:case:code[2]:19,6}{5}
\tracepos{0/60:case:code[1]:20,1}{31015874sp}{29476732sp}{square}
\tracepos{0/60:case:code[1]:20,1}{31721294sp}{28771312sp}{square}
\boxchar{0/60:case:code[1]:20,1}{0}
\tracepos{0/60:case:code[1]:20,2}{31015874sp}{28362620sp}{square}
\tracepos{0/60:case:code[1]:20,2}{31721294sp}{27657200sp}{square}
\boxchar{0/60:case:code[1]:20,2}{1}
\tracepos{0/60:case:code[1]:20,3}{31015874sp}{27248508sp}{square}
\tracepos{0/60:case:code[1]:20,3}{31721294sp}{26543088sp}{square}
\boxchar{0/60:case:code[1]:20,3}{2}
\tracepos{0/60:case:code[1]:20,4}{31015874sp}{26134396sp}{square}
\tracepos{0/60:case:code[1]:20,4}{31721294sp}{25428976sp}{square}
\boxchar{0/60:case:code[1]:20,4}{3}
\tracepos{0/60:case:code[1]:20,5}{31015874sp}{25020284sp}{square}
\tracepos{0/60:case:code[1]:20,5}{31721294sp}{24314864sp}{square}

```

```
\boxchar{0/60:case:code[1]:20,5}{4}  
\tracepos{0/60:case:code[1]:20,6}{31015874sp}{23906172sp}{square}  
\tracepos{0/60:case:code[1]:20,6}{31721294sp}{23200752sp}{square}  
\boxchar{0/60:case:code[1]:20,6}{5}
```

Contents

1	Introduction	1
2	Samples	1
2.1	Standard layout	4
2.2	Separate answer sheet	5
2.3	Without markers	6
3	Usage	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	21
3.9.1	Boxes	21
3.9.2	Codes	22
3.9.3	Answers	22
4	Implementation	23
4.1	Variables	23
4.2	Dimensions	26
4.3	Human readable sheet ID position	26
4.4	Localisation	27
4.4.1	English	27
4.4.2	Catalan	28
4.4.3	Dutch	28
4.4.4	French	28
4.4.5	German	29
4.4.6	Italian	29
4.4.7	Norwegian	29
4.4.8	Portuguese	30
4.4.9	Spanish	30
4.4.10	Japanese	30
4.4.11	Other languages	31
4.5	Interaction with other packages	31
4.5.1	cleveref	31
4.6	Random	31
4.6.1	Random pseudo-generator	31
4.6.2	Uniform random deviates	31
4.6.3	Tokens shuffling	32
4.7	Keys numbering	32
4.8	Boxes	33
4.8.1	Character logging	33
4.8.2	Position logging	33
4.8.3	Boxes to be checked by students	35

4.8.4	Scoring zones	40
4.8.5	Binary boxes	41
4.9	Checking Environment	42
4.10	Handling groups of questions	43
4.11	Questions	47
4.11.1	Managing answers	47
4.11.2	Separate answer sheet	48
4.11.3	Formatting answers	53
4.11.4	Score zones	55
4.11.5	Formatting questions	57
4.11.6	Explanations	59
4.12	Scoring	60
4.13	Numerical data	60
4.13.1	Codes	60
4.13.2	Numerical questions	64
4.13.3	Intervals	78
4.14	Open questions	78
4.15	Boxes with letters only	81
4.16	Page formatting	82
4.16.1	Watermark	82
4.16.2	Signs for scan analysis	83
4.17	Defining a single exam copy content	84
4.18	Pre-association	86
4.19	Package options	86
4.20	Survey add-on	88
4.21	Package Errors	89
4.22	Optional features	89
4.23	Use with recent LuaTeX versions	90
4.24	External control	90
4.25	Page layout	91
4.26	Initialisation	91
4.27	French command names	92
5	Outputs	92
5.1	namefield command	92
5.2	AMCboxedchar command	93
5.3	AMCcode command	93

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			1482, 1483, 1485,	1898, 1904, 1908,
\"	150, 156	1486, 1488, 1489,	1909, 1911, 1914,	
\@aucune	843, 847, 848	1490, 1491, 1511,	1915, 1922, 1932,	
\@firstoftwo	661	1512, 1517, 1518,	1933, 1934, 1935,	
\@ifl@t@r	3, 2361	1519, 1520, 1521,	1936, 1937, 1938,	
\@ifstar	1080, 1081	1522, 1527, 1529,	1939, 1940, 1941,	
\@secondoftwo	663	1530, 1531, 1532,	1943, 1948, 1949,	
\@skiphyperreffalse	1262	1533, 1535, 1537,	1952, 1958, 1960,	
\@skiphyperreftrue	1258, 1262	1538, 1539, 1540,	1962, 1963, 1966,	
\@tempboxa	2328, 2332, 2333	1543, 1550, 1551,	1967, 1971, 1972,	
\@tempdimb	2317, 2319	1552, 1554, 1556,	1973, 1975, 1976,	
\@tempdimc	2318, 2319	1557, 1559, 1561,	1978, 1979, 1980,	
_	1298	1563, 1566, 1567,	1981, 1982, 1984,	
\~	191, 192, 197, 198	1569, 1571, 1572,	1985, 1988, 1990,	
		1573, 1575, 1581,	1995, 1996, 1997,	
		1586, 1598, 1632,	1998, 2000, 2001,	
A			2002, 2003, 2004,	
\aa	177	1634, 1636, 1638,	2005, 2006, 2009,	
\AddToHook	2362	1640, 1643, 1646,	2010, 2011, 2012,	
\alafin	21, 2732	1649, 1652, 1655,	2016, 2018, 2020,	
\amc	870, 881,	1658, 1661, 1664,	2023, 2028, 2029,	
	883, 885, 886, 888,	1667, 1669, 1671,	2030, 2031, 2032,	
	891, 892, 894, 897,	1674, 1683, 1685,	2033, 2034, 2036,	
	898, 901, 902, 903,	1693, 1697, 1698,	2037, 2038, 2039,	
	904, 906, 908, 909,	1699, 1705, 1707,	2043, 2044, 2046,	
	911, 913, 914, 916,	1719, 1720, 1721,	2052, 2059, 2064,	
	918, 919, 922, 923,	1722, 1723, 1724,	2066, 2067, 2073,	
	925, 926, 928, 929,	1725, 1726, 1729,	2082, 2084, 2085,	
	931, 932, 934, 935,	1732, 1736, 1739,	2086, 2087, 2091,	
	937, 940, 941, 943,	1742, 1746, 1749,	2092, 2093, 2097,	
	945, 949, 950, 951,	1753, 1756, 1761,	2098, 2099, 2100,	
	952, 1363, 1364, 1365,	1763, 1765, 1772,	2102, 2103, 2104,	
	1366, 1367, 1368,	1774, 1777, 1786,	2117, 2119, 2125,	
	1369, 1371, 1378,	1787, 1788, 1789,	2134, 2137, 2138,	
	1381, 1386, 1390,	1791, 1793, 1796,	2144, 2145, 2146,	
	1398, 1401, 1402,	1801, 1802, 1804,	2147, 2148, 2149,	
	1403, 1406, 1411,	1808, 1810, 1813,	2150, 2152, 2153, 2156	
	1413, 1416, 1417,	1814, 1815, 1816,	\AMC@addpages	2445, 2447, 2449
	1418, 1419, 1421,	1818, 1819, 1820,	\AMC@affecte	281, 1273
	1423, 1425, 1426,	1823, 1826, 1831,	\AMC@amclog	19
	1428, 1429, 1433,	1833, 1836, 1840,	\AMC@answerBox	480
	1434, 1435, 1437,	1841, 1842, 1844,	\AMC@answerBox@	361,
	1438, 1439, 1441,	1847, 1852, 1858,		488, 535, 537, 553,
	1446, 1448, 1450,	1860, 1862, 1865,		555, 621, 623, 631, 633
	1458, 1463, 1464,	1873, 1874, 1881,	\AMC@binaryBoxes	654
	1465, 1467, 1469,	1882, 1884, 1885,	\AMC@binaryCode	604,
	1472, 1473, 1474,	1886, 1888, 1891,		655, 2348, 2349, 2350
	1475, 1477, 1479,	1892, 1895, 1896,		

\AMC@box	522, 1142, 1143, 1147, 2589, 2590	\AMC@keepmemoryfalse . .	2435	\AMC@logfile .	19, 20, 21, 290
\AMC@boxcolor 379, 505, 2239		\AMC@keepmemorytrue . . .	997	\AMC@LR	23, 392, 2320, 2371, 2372, 2373, 2374, 2375, 2690
\AMC@boxcolor@	379, 380, 381, 383, 401, 410, 422, 426, 443	\AMC@keyBox@	551, 1389, 1712, 1715	\AMC@makeovalbox	415, 434, 435, 436, 437
\AMC@boxeddown 393, 493, 504		\AMC@keys@line	2512	\AMC@mem@add	911, 957, 964, 971, 978, 1497
\AMC@boxedheight	406, 420, 421, 423, 424, 427, 428, 472, 495, 499, 500, 512, 515	\AMC@keys@next	2511	\AMC@mem@add@ifneeded . .	868, 1305
\AMC@boxedrule	398, 420, 421, 422, 491, 502	\AMC@lang@code	2480, 2515, 2516, 2712	\AMC@mem@addsingle@ifneeded	960, 1073, 1075, 1077, 1079
\AMC@boxedwidth 409, 420, 421, 423, 424, 427, 428, 472, 494, 499, 501, 512, 513, 1043, 1809		\AMC@loc@CA	107	\AMC@mem@addvar	916
\AMC@catalogMode	2505, 2513, 2672	\AMC@loc@catalog 101, 114, 128, 141, 155, 169, 182, 194, 208, 220, 2510		\AMC@mem@aid	935, 970
\AMC@CBtaille 602, 2376, 2519		\AMC@loc@corrected	100, 113, 127, 140, 154, 168, 181, 193, 207, 219, 2344	\AMC@mem@answer	967, 1107, 1114, 1115, 1116, 1121, 1127
\AMC@checkbox	361, 382, 385, 435, 437, 447, 453, 463, 621, 623, 1142, 1710, 1730, 1742, 1746, 2589	\AMC@loc@DE	148	\AMC@mem@category .	929, 963, 2129, 2275, 2311
\AMC@chiffres	1351	\AMC@loc@draft . . 95, 108, 122, 135, 149, 163, 176, 189, 202, 215, 2315		\AMC@mem@clear . . . 886, 2437	
\AMC@coli	1008, 1012, 1013, 1014	\AMC@loc@ES	201	\AMC@mem@next .	906, 962, 976
\AMC@crosschar	404, 507	\AMC@loc@explain 102, 115, 142, 156, 195, 221, 1337		\AMC@mem@openQuestion . .	974, 1304
\AMC@crossrule 426, 492, 508		\AMC@loc@FR . . 134, 2515, 2727		\AMC@mem@qidaffname 923, 977	
\AMC@definitnumero . 282, 286		\AMC@loc@IT	162	\AMC@mem@show .	950, 987, 996
\AMC@draw@crossfalse . . 384		\AMC@loc@JA	214	\AMC@mem@show@filter 951, 991	
\AMC@draw@crosstrue . . . 386		\AMC@loc@message 96, 109, 123, 136, 150, 164, 177, 190, 203, 216, 2690		\AMC@mn@leftmargin	1153, 1161, 1164, 1170
\AMC@error@explain	1337, 1339, 2615	\AMC@loc@namesurname . . .	106, 119, 132, 146, 160, 173, 186, 199, 212	\AMC@mn@rightmargin	1154, 1162, 1165, 1168
\AMC@fillcolor@	382, 400, 401, 417, 422	\AMC@loc@NL	121	\AMC@mn@sep .	1152, 1168, 1170
\AMC@fin@rep . . 830, 1092, 1096, 1100, 1103, 1106		\AMC@loc@NO	175	\AMC@mn@test 1151, 1158, 1167	
\AMC@formBox	522	\AMC@loc@none . . 103, 116, 129, 143, 157, 170, 183, 196, 209, 222, 843		\AMC@multi@report 1416, 2443	
\AMC@formBox@	522	\AMC@loc@PT	188	\AMC@multiclear .	1421, 2434
\AMC@formcolspecs 1009, 1063		\AMC@loc@q	99, 112, 126, 139, 153, 167, 180, 192, 206, 218, 1265	\AMC@NCBcheck 598, 2350, 2519	
\AMC@formspecs . . 1000, 1015		\AMC@loc@qf	98, 111, 125, 138, 152, 166, 179, 191, 205, 217, 852	\AMC@NCBetud . 598, 2348, 2518	
\AMC@fullGroupsfalse . . . 36		\AMC@loc@question	104, 117, 130, 144, 158, 171, 184, 197, 210, 223, 229	\AMC@NCBpage . 598, 2349, 2518	
\AMC@fullGroupstrue	2483, 2509	\AMC@loc@questions	105, 118, 131, 145, 159, 172, 185, 198, 211, 224, 229	\AMC@new@savebox	362, 2544, 2548, 2549, 2550, 2551
\AMC@if@separate@question	881, 956, 961, 969, 975, 1496, 2129, 2275, 2311	\AMC@logchar	289, 390	\AMC@note	2345, 2346, 2384, 2394, 2689
\AMC@imax . 715, 727, 728, 729				\AMC@numeric@alsocorrect	1689, 2055, 2059, 2069, 2073
\AMC@intituleHead	2344, 2365, 2510			\AMC@numeric@keepas	1687, 2048, 2049
				\AMC@numeric@scoreapprox	1679, 2014, 2024
				\AMC@numeric@scoreexact	1677, 2013, 2019

\AMC@numeric@scorewrong	\AMC@smashboxheight	\AMC@sza@callout
..... 1681, 2021, 2024	. 367, 370, 371, 372, 375	1230, 1235, 1239, 1241
\AMC@numeration	\AMC@smashcentered	\AMC@sza@callout@margin
..... 281, 285, 286	. 368, 410, 411, 443, 444	\AMC@sza@callout@margin
\AMC@outside@sep	\AMC@SR 1224
.. 497, 503 231,	\AMC@sza@callout@none
\AMC@oval@radius	233, 236, 237, 238,	. 1212
..... 422, 496, 513, 515	239, 241, 243, 244, 2428	\AMC@sza@callout@question
\AMC@ovalbox@	\AMC@SR@count 1216
436, 450, 2550 236,	\AMC@sza@depth
\AMC@ovalbox@F	237, 239, 247, 251, 1209, 1229, 1234
437, 448, 2551	254, 257, 258, 259, 260	\AMC@sza@height
\AMC@ovalbox@R	\AMC@SR@time 1209, 1228, 1233
..... 433, 434, 456, 2548 248, 2496	\AMC@sza@init@margin
\AMC@ovalbox@RF	\AMC@SR@advance	.. 1219
435, 454, 2549	234, 241, 242	\AMC@sza@init@margin
\AMC@pageHook	\AMC@SR@bit	. 1223
2355, 2362, 2363 241	\AMC@sza@init@none
\AMC@pagepos	\AMC@SR@const 1211
..... 295, 2356	.. 232, 236, 239	\AMC@sza@init@question
\AMC@premierecopie	\AMC@SR@max	1215
..... 603, 2418, 2421, 2520 247, 275	\AMC@sza@width
\AMC@prepare	\AMC@SR@nextByte 1209, 1227, 1232
.. 284, 287, 288 247	\AMC@tempenv
\AMC@prepare@element	\AMC@SR@num 659, 660
..... 678, 686, 801	249, 250, 252,	\AMC@tf@colW
\AMC@printformoutside@false	253, 257, 260, 275, 276 1042, 1043, 1046, 1052
..... 526	\AMC@SR@set	\AMC@tf@idtext
\AMC@printformoutside@true	233, 245, 246, 248 1023, 1028, 1045, 1048
..... 528, 529	\AMC@SR@test 1001, 1024, 1029
\AMC@printkeyoutside@false 242, 253	\AMC@tf@idtitle
..... 543	\AMC@SR@value 1026, 1039, 1040, 1052, 1063
\AMC@printkeyoutside@true 244	\AMC@tf@ncols
..... 545, 547	\AMC@stepQuestion	1051, 1059, 1063, 1067
\AMC@qaff 1255, 1290, 1330	\AMC@tfaw
..... 853,	\AMC@sti 1022, 1055, 1057, 1058
1255, 1298, 1299, 1333	262, 270, 274, 277, 278	\AMC@tf@box
\AMC@save@box	\AMC@stil	.. 1021, 1052,
363, 418, 2545 263,	1053, 1054, 1057, 1058
\AMC@setcolors@	271, 272, 273, 275, 279	\AMC@tracebox
..... 378, 399, 416, 441	\AMC@surveyfalse 295, 349, 442, 468,
\AMC@shape@form 2529	586, 588, 591, 1123, 2338
..... 475	\AMC@surveytrue	\AMC@tracechar
\AMC@shape@form@base 2530	337, 539, 557
462, 476	\AMC@sz@box	\AMC@tracepos
\AMC@shape@form@ticked 1174, 296, 407, 413,
..... 464, 466, 469	1176, 1178, 1181, 1224	1109, 1112, 1296, 1305
\AMC@shape@none	\AMC@sz@callin	\AMC@traceposx
..... 479 1186, 1191, 1199, 305, 325
\AMC@shape@oval	1201, 1266, 1300, 1301	\AMC@traceposy
..... 440	\AMC@sz@callin@question	314, 324, 326
\AMC@shape@square	1176	\AMC@unnumero
..... 397	\AMC@sz@callout 281
\AMC@shapename 1185, 1190,	\AMC@use@box
..... 295, 303, 312, 321, 498	1194, 1196, 1293, 1294 364,
\AMC@shapename@	\AMC@sz@callout@margin	448, 450, 454, 456, 2546
295, 393, 509	1178	\AMC@VERSION
\AMC@shapeprepare	\AMC@sz@callout@margin 2704, 2706
..... 509, 517, 2552	1181	\AMC@watertext
\AMC@shapeprepare@form	\AMC@sz@depth	.. 2315, 2326
461	1174, 1184, 1189	\AMC@XYFILE
\AMC@shapeprepare@none	\AMC@sz@height	298, 307, 316,
478 1174, 1183, 1188	328, 333, 334, 335,
\AMC@shapeprepare@oval	\AMC@sz@init@margin	339, 345, 347, 1276,
432	.. 1180	1299, 1413, 2427,
\AMC@shapeprepare@square	\AMC@sz@width	2471, 2476, 2477,
396	1174, 1182, 1187	
\AMC@shuffletoks	\AMC@sza@box	
..... 262, 748, 751, 823	1209, 1217, 1220	
\AMC@smashbox	\AMC@sza@callin	
.. 366, 369,	862, 863,	
370, 371, 372, 373, 376	1231, 1236, 1244, 1246	
	\AMC@sza@callin@margin	
	1221	
	\AMC@sza@callin@margin	
	1225	
	\AMC@sza@callin@none	
	.. 1213	
	\AMC@sza@callin@question	
 1217	

2702, 2703, 2704, 2705, 2706, 2707, 2708, 2709, 2710, 2711, 2712, 2714, 2716	\AMCcodeGrid 15, 1351	\AMCgrouploop@next 769, 782, 800
\AMCaddpagesto 2445	\AMCcodeGridInt ... 15, 1351	\AMCgrouploop@prep 755, 780, 798
\AMCanswer 1127, 1128, 2228, 2290	\AMCcodeH 1491	\AMCgrouppre@cyclic ... 738
\AMCassociation 2469	\AMCcodeHspace 1352, 1393, 1452	\AMCgrouppre@fixed ... 700
\AMCbeforeQuestion 1264, 1292, 1373	\AMCcodeID 1357, 1432	\AMCgrouppre@withoutreplacement 716
\AMCbeginAnswer 1099, 1128, 2227, 2289	\AMCcodeID@@ 1356, 1360, 2528, 2678	\AMCgrouppre@withreplacement 707
\AMCbeginQuestion 1264, 1298, 1313, 1319, 1333, 1372	\AMCcodeID@dot 1355	\AMChorizAnswerSep 1117, 1118, 1126
\AMCbin@begin 653	\AMCcodeID@mode . 1358, 2705	\AMChorizBoxSep 1119, 1120, 1123, 1125
\AMCbin@did 608, 616, 619, 621, 626, 629, 631, 637	\AMCcodeID@squarebrackets 1354	\AMCid@check 29, 330, 2350, 2357, 2358, 2377, 2381, 2696
\AMCbin@digit 606, 639, 642, 648, 651	\AMCcodeVspace 1353, 1391, 1451, 1454	\AMCid@checkmax 598, 2358, 2518, 2696
\AMCbin@hsep 610, 613, 617, 627	\AMCcompleteMulti .. 12, 65	\AMCid@etud . 30, 292, 300, 309, 318, 330, 333, 334, 335, 340, 355, 476, 1299, 2347, 2348, 2377, 2381, 2389, 2417, 2418, 2420, 2428, 2433, 2440, 2441, 2461, 2462, 2471
\AMCbin@id 607, 612, 620, 621, 630, 631	\AMCcurrentenv 657, 660, 1289	\AMCid@etudfin 32, 2419, 2420, 2421, 2440
\AMCbin@ndigits 609, 611, 648, 651	\AMCdebutFormulaire 21, 2734	\AMCid@etudstart 31, 2417, 2441
\AMCbin@number 605, 638, 641, 643, 645	\AMCdecimalPoint 1620, 1805, 1837	\AMCid@name . 67, 567, 570, 572, 576, 578, 920, 977, 1109, 1112, 1123, 1273, 1276, 1305, 1389, 1712, 1715, 2171
\AMCbin@one 615, 643	\AMCdefault@groupmode 675, 689, 690	\AMCid@quest 28, 333, 334, 335, 567, 570, 572, 576, 578, 586, 588, 591, 919, 977, 1109, 1112, 1123, 1273, 1275, 1276, 1296, 1299, 1305, 1389, 1713, 1716, 1945, 2210, 2286
\AMCbin@sequence .. 604, 638, 643, 644, 649, 650	\AMCdontAnnotate . 333, 2244	\AMCIDBoxesA 2348, 2352
\AMCbin@style 614, 638	\AMCdontScan 333, 2242	\AMCIDBoxesABC .. 2351, 2378
\AMCbin@zero .. 625, 644, 649	\AMCdump@reponses . 823, 837	\AMCIDBoxesB 2349, 2353
\AMCbloc 1271	\AMCemptybox 1149, 1174, 1209	\AMCIDBoxesC 2350, 2353
\AMCbo@help . 2279, 2280, 2293	\AMCendAnswer 1100, 1128, 2227, 2289	\AMCids@height . 79, 90, 2377
\AMCbo@hide 2306, 2312	\AMCexponent 1622, 2089, 2101	\AMCids@sidefalse ... 82, 84
\AMCbo@opts 2283	\AMCexternalQuestion .. 1280	\AMCids@sidetrue 86
\AMCbo@show 2284, 2312	\AMCform . 16, 868, 1069, 2733	\AMCids@topfalse 82, 86
\AMCbotextGoto .. 2278, 2307	\AMCformAfterQuestion 851, 1305	
\AMCboxColor 520, 2376	\AMCformAnswer 850, 1005, 1046	
\AMCboxDimensions . 521, 638	\AMCformAnswerA ... 867, 971	
\AMCboxedAnswers 1082, 2487	\AMCformatChoices 1494, 2130, 2276, 2312	
\AMCboxOnly 2278	\AMCformBeforeQuestion 850, 856, 1044	
\AMCboxOutsideLetter 522, 2229, 2291	\AMCformBegin . 16, 868, 2734	
\AMCboxStyle 21, 491, 2487, 2537	\AMCformFilter 989	
\AMCccircle 2334, 2337	\AMCformHSpace .. 68, 866, 1001, 1042, 1045, 2290	
\AMCchoiceLabel 480, 561, 1005, 1068	\AMCformQuestion . 850, 1045	
\AMCchoiceLabelFormat 394, 480	\AMCformQuestionA . 854, 978	
\AMCcleardoublepage 16, 2454	\AMCformQuestionN . 853, 861	
\AMCcode 1490	\AMCformS 868	
	\AMCformTH 1001, 1020, 1030, 1045, 1047, 1049, 1053	
	\AMCformVSpace 68, 850, 1044, 1064	
	\AMCgroup@pre 744, 767	

\AMCids@toptrue	84	\AMCneeds@storeboxfalse	2525	\AMCopen@LineHeight	2178, 2179, 2200
\AMCids@width	78, 89, 2380	\AMCneeds@storeboxtrue	2526	\AMCopen@Lines	2180, 2181, 2205
\AMCidsPosition	76	\AMCnobloc	1270, 1377	\AMCopen@lines	2198
\AMCidsVar	80	\AMCnoCompleteMulti	12, 66, 2226	\AMCopen@lineuptext	2167, 2190, 2215, 2218, 2221
\AMCidsVarN	80, 81	\AMCnombreCopies	2413, 2414, 2713, 2716	\AMCopen@question	2166, 2171, 2234, 2235
\AMCif@env	658, 1337, 1339	\AMCnoScoreZone	1254, 1374	\AMCopen@Width	2176, 2177, 2199
\AMCifcategory	952	\AMCnntextGoto	1611, 2121	\AMCopenHide	2270, 2276
\AMCinterBquest	73, 1305	\AMCnntextSign	1618, 1856, 1857	\AMCopenOpts	2197
\AMCinterBrep	68, 1113	\AMCnntextVHead	1612, 1809, 1955	\AMCopenShow	2208, 2276
\AMCinterIquest	72, 1305	\AMCnum@copies	33, 2413, 2414, 2415, 2416, 2419	\AMCotextGoto	2159, 2271
\AMCinterIrep	68, 1090	\AMCnum@questions	824, 826	\AMCotextReserved	2160, 2170, 2246, 2250, 2252
\AMCintervalFormat	2141, 2152	\AMCnumeric@Hspace	1613, 1625, 1767, 1769, 1805, 1824, 2101	\AMCoutsideLabelFormat	522
\AMCintervals	17, 2141, 2736	\AMCnumeric@Vspace	1614, 1624, 1740, 1747, 1754, 1781, 1797, 1811, 1845, 1857, 2088, 2090	\AMCpageref	2465
\AMClabel	2462, 2467	\AMCnumericChoices	18, 1494	\AMCpostNquest	74, 2110
\AMCload@reponse	834, 836, 1142, 1147, 2589	\AMCnumericChoicesPlain	2128, 2135	\AMCpostOquest	75, 2263
\AMCload@counter	27, 812, 813, 814, 815, 818, 819, 823, 826, 827, 828, 829, 832	\AMCnumericHide	2125, 2130	\AMCqlabel	2466
\AMCload@reponse	817, 836	\AMCnumericOpts	1705	\AMCquestionaff	1255
\AMClocalized	94	\AMCnumericShow	2117, 2130	\AMCquestionNumberfalse	1375
\AMCloop@k	721, 727, 731, 733, 754, 762, 771	\AMCnumero	1256, 2432	\AMCquestionNumbertrue	47
\AMCmarginNote	1155, 1181, 1224	\AMCcocol@Background	2161, 2168, 2233	\AMCrandomseed	245
\AMCmem@elt@cat	926, 938, 942	\AMCcocol@BoxFrameRule	2162, 2186, 2233	\AMCref	2462, 2462
\AMCmessage	19, 282, 594, 996, 1140, 1146, 1275, 1283, 1284, 1285, 1305, 1309, 1314, 1320, 1347, 1348, 1349, 1350, 1388, 1502, 1709, 1741, 1743, 1748, 1750, 1755, 1757, 1766, 1779, 1795, 1887, 1890, 1923, 1925, 1945, 2010, 2017, 2034, 2036, 2039, 2041, 2044, 2046, 2049, 2053, 2056, 2058, 2065, 2070, 2072, 2112, 2210, 2267, 2286, 2303, 2415, 2433, 2536, 2540, 2587, 2697, 2698, 2699, 2700	\AMCcocol@Foreground	2164, 2169, 2238, 2239, 2249, 2252	\AMCrep@count	838, 840, 842
\AMCncol@Background	1630, 2081	\AMCcocol@FrameRule	2163, 2187, 2260	\AMCrep@bloc	1089, 1107
\AMCncol@Border	1626, 2081	\AMCOpen	2159	\AMCrep@count	567, 570, 572, 576, 578, 809, 820, 840, 842, 846, 932, 970, 1084, 1109, 1112, 1123, 1132, 1140, 1143, 1146, 1147, 1383, 1387, 1388, 1389, 1495, 1497, 1708, 1709, 1713, 1716, 2586, 2587, 2590
\AMCncol@BorderWidth	1628, 2080	\AMCopen@answer	2165, 2172, 2201, 2217	\AMCrep@fini	834, 837, 841, 849
		\AMCopen@boxframerule	2184, 2185, 2232	\AMCrep@init	830, 1091, 1094, 1098, 1102, 1105
		\AMCopen@boxmargin	2182, 2183, 2231	\AMCrep@itemize	1088, 1107
		\AMCopen@contentcommand	2173, 2261	\AMCrep@lign	1094, 1107
		\AMCopen@framerule	2188, 2189, 2259	\AMCrep@nn	810, 1085, 1133, 1134, 1135
		\AMCopen@Hspace	2174, 2175, 2230, 2235, 2247	\AMCrep@nnmax	811, 1040, 1134, 1135, 2426
				\AMCrep@o	831, 833
				\AMCrep@perso	1098, 1107
				\AMCrep@r	835
				\AMCrep@tikz	1102, 1107

\ding	470				
\dontannotate	334				
\dontscan	333				
\dotfill	358, 2203, 2219				
\draw ...	422, 426, 1168, 1170				
E					
\element	14, 666				
\endchoices	2719				
\endchoicescustom	2721				
\endchoiceshoriz	2720				
\endcopieexamen	2729				
\endexamcopy	2729				
\endreponses	2719				
\endreponseshoriz	2720				
\endreponsesperso	2721				
ensemble (option)	21				
environments:					
amcxyfile	344				
auto	2554				
choices	12, 1082				
choicescustom ..	12, 1082				
choiceshoriz ...	12, 1082				
copieexamen	21				
examcopy	2626				
question	11, 1269				
question-auto	2554				
questionmult ...	11, 1269				
Questionnaires	2554				
questionouverte ...	1269				
reponses	21				
reponseshoriz	21				
reponsesperso	21				
tikz-multi	1082				
tikz-single	1082				
values	2554				
values-auto	2554				
variable-auto	2554				
variable-multi	1269				
variable-single ...	1269				
\evensidemargin .	1162, 1164				
\examcopy	2729				
examcopy (env.)	2626				
\exemplaire	21, 2727				
\exemplairepair	2461				
\explain	1335, 2616				
\explaincontext	1342				
\ExplSyntaxOff					
... 954, 1493, 1610,					
2127, 2140, 2158, 2653					
\ExplSyntaxOn					
... 868, 1361, 1507,					
1615, 2132, 2142, 2634					
F					
\fancygostyle					
... 2364, 2369, 2391, 2398					
\fbox	2389				
\fmtversion	3, 2361				
\footrulewidth					
... 2367, 2386, 2396, 2401					
\foreach	2580				
\formulaire	21, 2733				
\fp	1521, 1529,				
1531, 1533, 1537,					
1542, 1544, 1546,					
1550, 1557, 1562,					
1569, 1587, 1590,					
1932, 1933, 1971,					
1972, 1981, 2009,					
2010, 2036, 2144,					
2145, 2147, 2148,					
2149, 2150, 2152, 2153					
francais (option)	10				
fullgroups (option)	11				
G					
\group	1513, 1525				
H					
\he@dbas ...	2342, 2371, 2372				
\he@dhaut	2343, 2375				
\he@dtaille .	2339, 2342, 2343				
\hfuzz	372				
\ht	370, 1150				
I					
\ifAMC@affichekeys					
... 38, 1297, 1298					
\ifAMC@asqbloc .	44, 851, 857				
\ifAMC@automarks					
... 62, 983, 2431, 2450, 2457					
\ifAMC@calibration					
... 48, 297, 306,					
315, 328, 333, 334,					
335, 338, 1140, 1146,					
1274, 1282, 1299,					
1308, 1347, 1348,					
1349, 1350, 1388,					
1412, 2427, 2433,					
2470, 2475, 2587, 2701					
\ifAMC@catalog	49, 389				
\ifAMC@correc	40,				
381, 1142, 1336, 1343,					
1711, 2201, 2217, 2589					
\ifAMC@correchead					
... 37, 2359,					
2405, 2421, 2682, 2688					
\ifAMC@draw@cross					
... 365, 404, 425					
\ifAMC@ensemble					
... 56, 527, 544,					
565, 584, 871, 983,					
986, 990, 995, 1298,					
1500, 1944, 2110,					
2111, 2209, 2214,					
2228, 2234, 2266,					
2285, 2290, 2293,					
2302, 2431, 2697, 2707					
\ifAMC@extractOnly					
... 42, 2078, 2711					
\ifAMC@fullGroups ..	36, 757				
\ifAMC@inside@box					
... 54, 547, 575,					
2229, 2291, 2698, 2708					
\ifAMC@inside@digit .	57, 481				
\ifAMC@invisible ...	63, 295				
\ifAMC@keepmemory	993, 2437				
\ifAMC@keyline	39, 1297, 1298				
\ifAMC@ordre	34, 831				
\ifAMC@outside@box					
... 55, 527, 545, 2699, 2709					
\ifAMC@pagelayout					
... 60, 2357, 2404, 2681					
\ifAMC@pdfform					
... 64, 353, 2424, 2442, 2535					
\ifAMC@plain	50, 2618				
\ifAMC@postcorrect					
... 61, 848, 2700, 2710					
\ifAMC@printformoutside					
... 525, 534					
\ifAMC@printformoutside@					
... 524, 531					
\ifAMC@printkeyoutside .					
... 542, 552					
\ifAMC@printkeyoutside@					
... 541, 549					
\ifAMC@qbloc ...	43, 1264,				
1296, 1305, 1332,					
1334, 2122, 2272, 2308					
\ifAMC@rbloc .	45, 1089, 1092				
\ifAMC@shuffleG	35, 748, 751				
\ifAMC@survey ...	2529, 2554				
\ifAMC@textPos .	41, 1109,				
1112, 1122, 1296, 1305					
\ifAMC@watermark					
... 53, 2359, 2687					
\ifAMC@zoneformulaire ..					
... 59, 529, 566, 872					
\ifAMC@complete@multi	46, 845				

