

The automultiplechoice package*

Alexis Bienvenue
paamc@passoire.fr

August 30, 2026

Abstract

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

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

1 Introduction

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

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

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

2 Samples

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

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

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

*This document corresponds to version revision: `r:5e5b2c22` from AMC 1.7.0+git20260830142431

```

        \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}

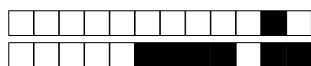
}
\end{document}
```

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

```

\usepackage[nopage,indivanswers]{automultiplechoice}

```

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

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

●  ● +1/1/60+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Egypt?

- ☐ Cayenne
- ☐ Caracas
- ☐ Cuito
- ☐ Conakry
- ☐ Chisinau

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

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

Question 4 What is the capital of Ghana?

☐ Accra ☐ Addis Ababa ☐ Ankara ☐ Apia

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

●  ● +2/1/60+

AMC SAMPLE TEST

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

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

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

- ☐ The first line of the Paris Métro 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
- ☐ 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-on becomes emperor of China

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

●  ● +3/1/52+

AMC SAMPLE TEST

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

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

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

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

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

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

Question 3 What is the capital of Ireland?

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

Question 4 What is the capital of Thailand?

☐ Beijing ☐ Bangor ☐ Bangkok ☐ Beirut ☐ Berlin

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

●  ● +4/1/52+

AMC SAMPLE TEST

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

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

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

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

Question 2 What is the capital of Egypt?

- ☐ Caracas
- ☐ Cayenne
- ☐ Cuito
- ☐ Conakry
- ☐ Chisinau

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

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

Question 4 What is the capital of Ireland?

- ☐ Djibouti ☐ Dhaka ☐ Dakar
- ☐ Dili ☐ Dhaka ☐ 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 Cmastry
☐ 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`. •

•  • +2/1/52+

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 Cete
☐ 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`. •

•  • +2/2/52+

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
☐ Cusackey
☐ Chisnaus

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

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

Question 4 What is the capital of Ghana?

☒ Accra ☐ Addis Ababa ☐ Ankara ☐ Apia

1

AMC

SAMPLE TEST

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

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

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

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

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

Question 2 What is the capital of Iceland?

☐ Djibouti ☐ Dhaka ☐ Doha
☒ Dshila ☐ Dili ☐ Dakar

Question 3 What is the capital of Ghana?

☐ Apia ☒ Accra ☐ Addis Ababa ☐ Ankara

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

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

1

AMC

SAMPLE TEST

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

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

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

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

☐ The first commercial Concorde flight takes off
☒ Apollo 14 lands on the Moon
☒ The Soviet Union launches 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-sun becomes emperor of China
☐ Napoleon Bonaparte crosses the Alps and invades Italy
☒ American Express is founded by Henry Wells & William Fargo

Question 3 What is the capital of Iceland?

☐ Dhaka ☐ Doha ☐ Dakar
☐ Dili ☒ Dshila ☐ 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
☐ Cusackey
☐ Chisnaus

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

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

Question 4 What is the capital of Iceland?

☐ Djibouti ☐ Dhaka ☐ Dakar
☐ Dili ☐ Doha ☒ Dshila

1

3 Usage

3.1 Package options

The following options are available for package `automultiplechoice`:

`noshuffle` cancels answers shuffling for all questions.

`noshufflegroups` cancels groups shuffling.

`answers` produces a common corrected answers sheet.

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

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

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

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

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

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

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

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

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

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

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

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

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

`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.

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

The name field

Name and surname:

.....

```

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

```

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

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

Student ID												
☐ 0	☐ 0	☐ 0	☐ 0	☐ 0	☐ 0	<code>\AMCcodeGridInt{student}{6}</code>						
☐ 1	☐ 1	☐ 1	☐ 1	☐ 1	☐ 1							
☐ 2	☐ 2	☐ 2	☐ 2	☐ 2	☐ 2							
☐ 3	☐ 3	☐ 3	☐ 3	☐ 3	☐ 3							
☐ 4	☐ 4	☐ 4	☐ 4	☐ 4	☐ 4							
☐ 5	☐ 5	☐ 5	☐ 5	☐ 5	☐ 5							
☐ 6	☐ 6	☐ 6	☐ 6	☐ 6	☐ 6							
☐ 7	☐ 7	☐ 7	☐ 7	☐ 7	☐ 7							
☐ 8	☐ 8	☐ 8	☐ 8	☐ 8	☐ 8							
☐ 9	☐ 9	☐ 9	☐ 9	☐ 9	☐ 9							

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

Student ID, horizontal form											
☐ 0	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5	☐ 6	☐ 7	☐ 8	☐ 9		
☐ 0	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5	☐ 6	☐ 7	☐ 8	☐ 9		
☐ 0	☐ 1	☐ 2	☐ 3	☐ 4	☐ 5	☐ 6	☐ 7	☐ 8	☐ 9		

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

3.6 Separate answer sheet

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

`\AMCform`

`\AMCcleardoublepage`

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

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

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

See section 2.2 for an example.

3.7 Random computation questions

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

Random computation questions

Question 6 How much are 2 plus 8?

☐ 9 ☒ 10 ☐ 16 ☐ -6

```

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

```

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

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

Pick the right interval

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

- | | | | | |
|-------------------------------------|--|-------------------------------------|-------------------------------------|-------------------------------------|
| ☐ [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[|

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

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

Numeric choices

Question 8

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

	☐ 0	☐ 1	☐ 2	☒ 3	☐ 4	☐ 5	☐ 6	☐ 7	☐ 8	☐ 9
	.									
☒ +	☐ 0	☐ 1	☐ 2	☒ 3	☐ 4	☐ 5	☐ 6	☐ 7	☐ 8	☐ 9
☐ -	☐ 0	☐ 1	☒ 2	☐ 3	☐ 4	☐ 5	☐ 6	☐ 7	☐ 8	☐ 9

```
\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 $\text{\LaTeX}$ package `tikz` must be loaded.

width for the width of the boxes.

height for the height of the boxes.

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

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

rule for the rule width.

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 $\text{\LaTeX}$ boxes (so that they can't be splitted on two different pages).

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

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

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

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

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

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

`\ifAMC@calibration` if this $\text{\LaTeX}$ run is used to get page layouts.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```
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@flat\AMC@flatfalse
51 \newif\ifAMC@plain\AMC@plainfalse
52 \newif\ifAMC@bonne
53 \newif\ifAMC@type@multi
54 \newif\ifAMC@watermark\AMC@watermarktrue
55 \newif\ifAMC@inside@box\AMC@inside@boxfalse
56 \newif\ifAMC@outside@box\AMC@outside@boxfalse
57 \newif\ifAMC@ensemble\AMC@ensemblefalse
58 \newif\ifAMC@inside@digit\AMC@inside@digitfalse
59 \newif\ifAMC@formulaire@dedans\AMC@formulaire@dedansfalse
60 \newif\ifAMC@zoneformulaire
61 \newif\ifAMC@pagelayout\AMC@pagelayouttrue
62 \newif\ifAMC@postcorrect\AMC@postcorrectfalse
63 \newif\ifAMC@automarks\AMC@automarksfalse
64 \newif\ifAMC@invisible\AMC@invisiblefalse
65 \newif\ifAMC@pdfform\AMC@pdfformfalse
```

```

66 \let\AMCcompleteMulti=\AMCcomplete@multitrue
67 \let\AMCnoCompleteMulti=\AMCcomplete@multifalse

```

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

```

68 \def\AMCid@name{}

```

4.2 Dimensions

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

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

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

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

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

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

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

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

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

```

69 \newdimen\AMCformVSpace\AMCformVSpace=1.2ex
70 \newdimen\AMCformHSpace\AMCformHSpace=.3em
71 \newdimen\AMCinterIrep\AMCinterIrep=\z@
72 \newdimen\AMCinterBrep\AMCinterBrep=.5ex
73 \newdimen\AMCinterIquest\AMCinterIquest=\z@
74 \newdimen\AMCinterBquest\AMCinterBquest=3ex
75 \newdimen\AMCpostNquest\AMCpostNquest=1.5ex
76 \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).

```

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

```

```

84 \or
85 \AMCids@toptrue\AMCids@sidefalse
86 \or
87 \AMCids@topfalse\AMCids@sidetrue
88 \fi
89 }
90 \define@key{AMCids}{width}{\AMCids@width=#1}
91 \define@key{AMCids}{height}{\AMCids@height=#1}
92 \def\AMCidsPosition#1{\setkeys{AMCids}{#1}}
93 \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.

```

94 \def\AMCtext#1#2{\expandafter\def\cname AMC@loc@#1\endcsname{#2}}
95 \def\AMClocalized#1{\cname AMC@loc@#1\endcsname}

```

4.4.1 English

Text indicating draft exams:

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

Message at page bottom when compiled out of AMC gui:

```

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

```

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

```
99 \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):

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

Headers for corrected version and catalog:

```

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

```

Localization text for Explanation

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

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

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

Word for 'question', singular and plural forms:

```

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

```

Default text to write in the students' name box:

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

4.4.2 Catalan

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

```
108 \def\AMC@loc@CA{
109   \def\AMC@loc@draft{PROJECTE}
110   \def\AMC@loc@message{Pel vostre examen, imprimiu preferiblement
111     els documents compilats amb l'ajuda de auto-multiple-choice.}
112   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
113   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
114   \def\AMC@loc@corrected{Correcci'o}
115   \def\AMC@loc@catalog{Cat'aleg}
116   \def\AMC@loc@explain{\textit{\textbf{Explicaci'o : }}}
117   \def\AMC@loc@none{Cap de les respostes 'es correcte.}
118   \def\AMC@loc@question{pregunta}
119   \def\AMC@loc@questions{preguntes}
120   \def\AMC@loc@namesurname{Nom i cognoms:}
121 }
```

4.4.3 Dutch

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

```
122 \def\AMC@loc@NL{
123   \def\AMC@loc@draft{Ontwerp}
124   \def\AMC@loc@message{Gebruik bij uw proefwerk bij voorkeur die
125     documenten welke door auto-multiple-choice zijn aangemaakt.}
126   \def\AMC@loc@qf##1{\textbf{Vraag ##1 :}}
127   \def\AMC@loc@q##1##2{\textbf{Vraag ##1} ##2}
128   \def\AMC@loc@corrected{Correctie}
129   \def\AMC@loc@catalog{Catalogus}
130   \def\AMC@loc@none{Geen van de antwoorden is juist.}
131   \def\AMC@loc@question{vraag}
132   \def\AMC@loc@questions{vragen}
133   \def\AMC@loc@namesurname{Achternaam en voornaam:}
134 }
```

4.4.4 French

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

```
135 \def\AMC@loc@FR{
136   \def\AMC@loc@draft{PROJET}
137   \def\AMC@loc@message{Pour votre examen, imprimez de pr'ef'ERENCE
138     les documents compil'es 'a l'aide de auto-multiple-choice.}
139   \def\AMC@loc@qf##1{\textbf{Question ##1 :}}
140   \def\AMC@loc@q##1##2{\textbf{Question ##1} ##2}
141   \def\AMC@loc@corrected{Correction}
142   \def\AMC@loc@catalog{Catalogue}
143   \def\AMC@loc@explain{\textit{\textbf{Explication : }}}
144   \def\AMC@loc@none{Aucune de ces r'eponses n'est correcte.}
145   \def\AMC@loc@question{question}
146   \def\AMC@loc@questions{questions}
147   \def\AMC@loc@namesurname{Nom et pr'enom :}
148 }
```

4.4.5 German

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

```
149 \def\AMC@loc@DE{
150   \def\AMC@loc@draft{ENTWURF}
151   \def\AMC@loc@message{Benutzen Sie f\"ur Ihre Pr\"ufung bevorzugt Dokumente die mit
152     auto-multiple-choice erstellt wurden.}
153   \def\AMC@loc@qf##1{\textbf{Frage ##1 :}}
154   \def\AMC@loc@q##1##2{\textbf{Frage ##1} ##2}
155   \def\AMC@loc@corrected{Korrektur}
156   \def\AMC@loc@catalog{Katalog}
157   \def\AMC@loc@explain{\textit{\textbf{Erkl\"arung : }}}
158   \def\AMC@loc@none{Keine dieser Antworten ist korrekt.}
159   \def\AMC@loc@question{Frage}
160   \def\AMC@loc@questions{Fragen}
161   \def\AMC@loc@namesurname{Vor- und Nachname:}
162 }
```

4.4.6 Italian

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

```
163 \def\AMC@loc@IT{
164   \def\AMC@loc@draft{BOZZA}
165   \def\AMC@loc@message{Per l'esame, \e preferibile stampare i documenti
166     a partire da auto-multiple-choice.}
167   \def\AMC@loc@qf##1{\textbf{Domanda ##1:}}
168   \def\AMC@loc@q##1##2{\textbf{Domanda ##1} ##2}
169   \def\AMC@loc@corrected{Correzione}
170   \def\AMC@loc@catalog{Catalogo}
171   \def\AMC@loc@none{Nessuna risposta \e giusta.}
172   \def\AMC@loc@question{domanda}
173   \def\AMC@loc@questions{domande}
174   \def\AMC@loc@namesurname{Nome e cognome:}
175 }
```

4.4.7 Norwegian

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

```
176 \def\AMC@loc@NO{
177   \def\AMC@loc@draft{UTKAST}
178   \def\AMC@loc@message{Det anbefales {\aa} skrive ut dokumentet
179     for gjennomgang \direkte fra auto-multiple-choice.}
180   \def\AMC@loc@qf##1{\textbf{Oppgave ##1 :}}
181   \def\AMC@loc@q##1##2{\textbf{Oppgave ##1} ##2}
182   \def\AMC@loc@corrected{Rettet}
183   \def\AMC@loc@catalog{Katalog}
184   \def\AMC@loc@none{Ingen svar er riktige.}
185   \def\AMC@loc@question{oppgave}
186   \def\AMC@loc@questions{oppgave}
187   \def\AMC@loc@namesurname{Etternavn og fornavn:}
188 }
```

4.4.8 Portuguese

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

```
189 \def\AMC@loc@PT{
190   \def\AMC@loc@draft{RASCUNHO}
191   \def\AMC@loc@message{Para o seu exame, use preferencialmente documentos compilados do auto-multiple-choice}
192   \def\AMC@loc@qf##1{\textbf{Quest\~ao ##1:}}
193   \def\AMC@loc@q##1##2{\textbf{Quest\~ao ##1} ##2}
194   \def\AMC@loc@corrected{Corrigido}
195   \def\AMC@loc@catalog{Cat\'alogo}
196   \def\AMC@loc@explain{\textit{\textbf{Justifique: }}}
197   \def\AMC@loc@none{Nenhuma das respostas apresentadas est\'a correta.}
198   \def\AMC@loc@question{Quest\~ao}
199   \def\AMC@loc@questions{Quest\~oes}
200   \def\AMC@loc@namesurname{Nome e apelido:}
201 }
```

4.4.9 Spanish

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

```
202 \def\AMC@loc@ES{
203   \def\AMC@loc@draft{BORRADOR}
204   \def\AMC@loc@message{Para revisi\'on, preferentemente imprimir documentos compilados
205     desde auto-multiple-choice.}
206   \def\AMC@loc@qf##1{\textbf{Pregunta ##1 :}}
207   \def\AMC@loc@q##1##2{\textbf{Pregunta ##1} ##2}
208   \def\AMC@loc@corrected{Correcci\'on}
209   \def\AMC@loc@catalog{Cat\'alogo}
210   \def\AMC@loc@none{Ninguna de estas preguntas son correctas.}
211   \def\AMC@loc@question{pregunta}
212   \def\AMC@loc@questions{preguntas}
213   \def\AMC@loc@namesurname{Nombre y apellidos:}
214 }
```

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).

```
215 \def\AMC@loc@JA{
216   \def\AMC@loc@draft{\diamond\diamond\diamond\diamond}
217   \def\AMC@loc@message{\diamond\diamond\diamond\diamondauto-multiple-choice\diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond\diamond}
218   \def\AMC@loc@qf##1{\textbf{\diamond##1:}}
219   \def\AMC@loc@q##1##2{\textbf{\diamond##1} ##2}
220   \def\AMC@loc@corrected{\diamond\diamond\diamond\diamond}
221   \def\AMC@loc@catalog{\diamond\diamond\diamond\diamond}
222   \def\AMC@loc@explain{\textit{\textbf{\diamond: }}}
223   \def\AMC@loc@none{\diamond\diamond\diamond\diamond}
224   \def\AMC@loc@question{\diamond}
225   \def\AMC@loc@questions{\diamond}
226 }
```

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:

```
227 \AtBeginDocument{\@ifpackageloaded{cleveref}{%
228   \message{AMC/cleveref integration loaded^^J}%
229   \crefalias{AMCquestionaff}{question}%
230   \crefname{question}{\AMC@loc@question}{\AMC@loc@questions}%
231 }{}}%
```

`\AMCrefs` In some cases, some questions can be used or not depending on the student copy. The macro `\AMCrefs{<macroname>}{<questions IDs list>}` selects the used questions before passing the list to the command specified as the first argument (for example `cref`, `cpageref` or other variants).

```
232 \ExplSyntaxOn
233
234 \seq_new:N \amc_refs_a_seq
235 \seq_new:N \amc_refs_b_seq
236
237 \cs_new:Npn \amc_refs_filter:n #1 {
238   \seq_set_from_clist:Nn \amc_refs_a_seq { #1 }
239   \seq_set_map:NNn \amc_refs_b_seq \amc_refs_a_seq { \AMCstudentlabel { ##1 } }
240   \seq_set_filter:NNn \amc_refs_a_seq \amc_refs_b_seq {
241     \cs_if_exist_p:c { r@##1 }
242   }
243   \seq_if_empty:NTF \amc_refs_a_seq { \seq_put_right:Nn \amc_refs_a_seq { NONE } } { }
244 }
245
246 \cs_new_eq:NN \AMCfilterRefs \amc_refs_filter:n
247
248 \cs_new:Npn \amc_func_refs:Nn #1 #2 {
249   \amc_refs_filter:n { #2 }
250   \exp_args:Nx #1 { \seq_use:Nn \amc_refs_a_seq { , } }
251 }
252
253 \cs_generate_variant:Nn \amc_func_refs:Nn { cn }
254
255 \cs_new_eq:NN \AMCrefs \amc_func_refs:cn
256
257 \ExplSyntaxOff
```

4.6 Random

4.6.1 Random pseudo-generator

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

```
258 \ifx\AMC@SR\undefined\newcount\AMC@SR\fi
259 \providecommand\AMC@SRconst{2097152}
260 \providecommand\AMC@SRset[i]{\global\AMC@SR#1 \ignorespaces}
261 \providecommand\AMC@SRadvance{%
```

```

262 \begingroup%
263 \ifnum\AMC@SR<\AMC@SRconst\relax\AMC@SR@count\z@\else\AMC@SR@count\@ne\fi%
264 \ifodd\AMC@SR\advance\AMC@SR@count\@ne\fi%
265 \global\divide\AMC@SR\tw%
266 \ifodd\AMC@SR@count\global\advance\AMC@SR\AMC@SRconst\relax\fi%
267 \endgroup}
268 \providecommand\AMC@SRbit{\AMC@SRadvance\ifodd\AMC@SR1\else0\fi}
269 \providecommand\AMC@SRtest[2]{\AMC@SRadvance%
270 \ifodd\AMC@SR#2\else#1\fi\ignorespaces}
271 \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>}`.

```

272 \AMC@SRset{1515}
273 \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.

```

274 \newcount\AMC@SR@count
275 \def\AMC@SR@time{\AMC@SRset{\time}}
276 \newcount\AMC@SRnum
277 \def\AMC@SRnextByte{\AMC@SRnum=\z@%
278 \AMC@SR@count=20%
279 \loop\multiply\AMC@SRnum\tw%
280 \AMC@SRtest{\advance\AMC@SRnum\@ne}{}%
281 \ifnum\AMC@SR@count>\@ne\advance\AMC@SR@count\m@ne\repeat%
282 }
283 \newcommand\AMC@SRmax[1]{\AMC@SRnextByte%
284 \AMC@SR@count=\AMC@SRnum%
285 \divide\AMC@SR@count by #1\relax%
286 \multiply\AMC@SR@count by #1\relax%
287 \advance\AMC@SRnum by -\AMC@SR@count%
288 }

```

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).

```

289 \newcount\AMC@sti
290 \newcount\AMC@stil
291 \newtoks\AMCsw@p@
292 \newcommand\AMCsw@p[2]{%
293 \global\AMCsw@p@=#1%
294 \global#1=#2%
295 \global#2=\AMCsw@p@}
296 \newcommand{\AMC@shuffletoks}[3][\@ne]{%

```

```

297 \AMC@sti=#2\relax%
298 \AMC@stil=#2\relax%
299 \advance\AMC@stil\@ne%
300 \advance\AMC@stil -#1\relax%
301 \@whilenum\AMC@sti>#1\do{%
302   \AMC@SRmax{\AMC@stil}\advance\AMC@SRnum #1\relax%
303   \AMCsw@p{\csname #3\romannumeral\AMC@SRnum\endcsname}%
304   {\csname #3\romannumeral\AMC@sti\endcsname}%
305   \advance\AMC@sti\m@ne\relax%
306   \advance\AMC@stil\m@ne\relax%
307 }}

```

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).

```

308 \newcount\AMC@numerotation\AMC@numerotation=\z@%
309 \def\AMC@definitnumero#1#2{\AMCmessage{NUM=#1=#2}}%
310 \expandafter\global\expandafter\def\csname AMC@numtab@#2\endcsname{#1}}
311 \def\AMC@prepare#1{\expandafter\ifx\csname AMC@numtab@#1\endcsname\relax%
312   \global\advance\AMC@numerotation\@ne%
313   \expandafter\AMC@definitnumero\expandafter{\the\AMC@numerotation}{#1}\fi}
314 \def\AMC@unnumero#1{\AMC@prepare{#1}\csname AMC@numtab@#1\endcsname}
315 \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.

```

316 \def\AMC@logchar#1#2{%
317   \protected@write\AMC@logfile}{%
318     \string\answer%
319     {\the\AMCid@etud/\thepage:#2}%
320     {#1}}%
321 }

```

4.8.2 Position logging

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

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

```

322 \def\AMC@shapename{\ifAMC@invisible none\else\AMC@shapename\fi}

```

```

323 \def\AMC@tracepos#1#2{%
324   \ifAMC@calibration\ifx\@empty#1\@empty\else%
325   \pdfsavepos\protected@write\AMC@XYFILE{}\{%
326     \string\tracepos%
327     {\the\AMCid@etud/\thepage:#2}%
328     {\noexpand\number\pdfastxpos sp}%
329     {\noexpand\number\pdfastypos sp}%
330     {\AMC@shapename}}%
331   \fi\fi}
332 \def\AMC@traceposx#1#2{%
333   \ifAMC@calibration\ifx\@empty#1\@empty\else%
334   \pdfsavepos\protected@write\AMC@XYFILE{}\{%
335     \string\tracepos%
336     {\the\AMCid@etud/\thepage:#2}%
337     {\noexpand\number\pdfastxpos sp}%
338     {0sp}%
339     {\AMC@shapename}}%
340   \fi\fi}
341 \def\AMC@traceposy#1#2{%
342   \ifAMC@calibration\ifx\@empty#1\@empty\else%
343   \pdfsavepos\protected@write\AMC@XYFILE{}\{%
344     \string\tracepos%
345     {\the\AMCid@etud/\thepage:#2}%
346     {0sp}%
347     {\noexpand\number\pdfastypos sp}%
348     {\AMC@shapename}}%
349   \fi\fi}
350 \newcommand\AMC@tracebox[3]{%
351   \vbox{\AMC@traceposy{#1}{#2}%
352     \hbox{\AMC@traceposx{#1}{#2}#3\AMC@traceposx{#1}{#2}}%
353     \AMC@traceposy{#1}{#2}}
354 \def\AMC@pagepos{%
355   \ifAMC@calibration\protected@write\AMC@XYFILE{}\{%
356     \string\page%
357     {\the\AMCid@etud/\thepage/\the\AMCid@check}%
358     {\the\paperwidth}{\the\paperheight}%
359     {\the\pdfpagewidth}{\the\pdfpageheight}}\fi}

```

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

```

\AMCReTick 360 \newcommand{\AMCdontScan}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontscan{\the\AMCid@etud,\thepage,\the\AMCid@check}}%
361 \newcommand{\AMCdontAnnotate}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\dontannotate{\the\AMCid@etud,\thepage,\the\AMCid@check}}%
362 \newcommand{\AMCReTick}{\ifAMC@calibration\immediate\write\AMC@XYFILE{\string\retick{\the\AMCid@etud,\thepage,\the\AMCid@check}}%
363 %

```

\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.

```

364 \newcommand\AMC@tracechar[4]{%
365   \ifAMC@calibration\ifx\@empty#3\@empty\else%
366     \protected@write\AMC@XYFILE{}\{%
367       \string\boxchar{\the\AMCid@etud/\thepage:#4}{#1}%
368     }%
369   \fi\fi%

```

370 }

`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.

```
371 \newenvironment{amcxyfile}[1]{%
372   \protected@write\AMC@XYFILE{}\string\xyopen{#1}}%
373 }{%
374   \protected@write\AMC@XYFILE{}\string\xyclose{}}%
375 }
```

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

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

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

```
377 \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:

```
378 \newcommand{\namefielddots}{%
379   \noindent%
380   \ifAMC@pdfform%
381     \hspace*{\fill}%
382     \TextField[name={\the\AMCid@etud:namefield},width=.95\linewidth,bordercolor=0 0 0]{}%
383     \hspace*{\fill}
384   \else%
385     \dotfill
386   \fi%
387 }
```

As an example,

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

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

produces the following box:

Name:

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

4.8.3 Boxes to be checked by students

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

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

Depending on the required shape for the boxes, the corresponding

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

command is used.

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

```
388 \def\AMC@checkbox{}
389 \let\AMC@new@savebox=\newsavebox
390 \let\AMC@save@box=\savebox
391 \let\AMC@use@box=\usebox
392 \newif\ifAMC@draw@cross
```

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

```
393 \newbox\AMC@smashbox
394 \newdimen\AMC@smashboxheight
395 \newcommand{\AMC@smashcentered}[1]{%
396   \setbox\AMC@smashbox\hbox{#1}%
397   \AMC@smashboxheight=\ht\AMC@smashbox%
398   \advance\AMC@smashboxheight by \dp\AMC@smashbox%
399   \vfuzz=\AMC@smashboxheight\hfuzz=\wd\AMC@smashbox%
400   \hspace*{-.5\wd\AMC@smashbox}\hbox to .5\wd\AMC@smashbox{%
401     \vbox to 0pt{%
402       \vspace*{-.5\AMC@smashboxheight}\vbox to .5\AMC@smashboxheight{%
403         \box\AMC@smashbox}}}%
404 }
```

`\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.

```
405 \newcommand\AMC@setcolors@[2]{%
406   \def\AMC@boxcolor@{\AMC@boxcolor}%
407   \ifx\@empty#1\@empty \def\AMC@boxcolor@{black}\fi%
408   \ifAMC@correc\def\AMC@boxcolor@{black}\fi%
409   \def\AMC@fillcolor@{\ifx #2\AMC@checkbox%
410     \AMC@boxcolor@\else white\fi}%
411   \AMC@draw@crossfalse%
412   \ifKV@AMC@dim@cross\ifx #2\AMC@checkbox%
```

```

413 \AMC@draw@crosstrue\fi\fi%
414 }
415 \newcommand\AMC@answerBox@[4]{%
416 \ifAMC@catalog%
417 \AMC@logchar{#1}{#4}%
418 \fi%
419 \AMC@LR{\hspace{0pt}}%
420 \lower\AMC@boxeddown\hbox{\csname AMC@shape@\AMC@shapename@\endcsname%
421 {AMC@choiceLabelFormat{#1}}{#2}{#3}{#4}}}%
422 }
423 \newcommand\AMC@shapeprepare@square{%
424 \newcommand\AMC@shape@square[4]{%
425 \fboxsep=\z@\fboxrule=\AMC@boxedrule%
426 \AMC@setcolors@{#3}{#2}%
427 \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
428 \fcolorbox{\AMC@boxcolor@}{\AMC@fillcolor@}%
429 {%
430 \boxput*(0,0){%
431 \ifAMC@draw@cross\AMC@crosschar\fi%
432 }{%
433 \vbox to \AMC@boxedheight{%
434 \AMC@tracepos{#3}{#4}%
435 \vfill%
436 \hbox to \AMC@boxedwidth{\hfill%
437 \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}%
438 \AMC@smashcentered{#2}%
439 \hfill}\vfill}}%
440 \AMC@tracepos{#3}{#4}}%
441 }

```

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

```

442 \newcommand\AMC@makeovalbox[3]{%
443 \AMC@setcolors@{#1}{#2}%
444 \ifKV@AMCdim@cross\def\AMC@fillcolor@{white}\fi%
445 \AMC@save@box{#3}{%
446 \begin{tikzpicture}%
447 \useasboundingbox (-0.5\AMC@boxedwidth-0.5\AMC@boxedrule,0.5\AMC@boxedheight+0.5\AMC@boxedrule)
448 rectangle (0.5\AMC@boxedwidth+0.5\AMC@boxedrule,-0.5\AMC@boxedheight-0.5\AMC@boxedrule);
449 \draw[\AMC@boxcolor@,fill=\AMC@fillcolor@,line width=\AMC@boxedrule,rounded corners=\AMC@oval@radi
450 (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight)
451 rectangle (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
452 \ifAMC@draw@cross
453 \draw[\AMC@boxcolor@,line width=\AMC@crossrule]
454 (-0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (0.5\AMC@boxedwidth,-0.5\AMC@boxedheight)
455 (0.5\AMC@boxedwidth,0.5\AMC@boxedheight) -- (-0.5\AMC@boxedwidth,-0.5\AMC@boxedheight);
456 \fi
457 \end{tikzpicture}}%
458 }
459 \newcommand\AMC@shapeprepare@oval{%
460 \ifx\AMC@ovalbox@R\undefined\else%
461 \AMC@makeovalbox{1}{-}{\AMC@ovalbox@R}%
462 \AMC@makeovalbox{1}{-}{\AMC@checkedbox}{\AMC@ovalbox@RF}%
463 \AMC@makeovalbox{-}{-}{\AMC@ovalbox@}%
464 \AMC@makeovalbox{-}{-}{\AMC@checkedbox}{\AMC@ovalbox@F}%

```

```

465 \fi%
466 }
467 \newcommand\AMC@shape@oval[4]{%
468 \AMC@setcolors@{#3}{#2}%
469 \AMC@tracebox{#3}{#4}{\boxput*(0,0){%
470 \AMC@smashcentered{\textcolor{\AMC@boxcolor@}{#1}}}%
471 \AMC@smashcentered{#2}%
472 }{%
473 \ifx\@empty#3\@empty%
474 \ifx #2\AMC@checkbox%
475 \AMC@use@box{\AMC@ovalbox@F}%
476 \else%
477 \AMC@use@box{\AMC@ovalbox@}%
478 \fi%
479 \else%
480 \ifx #2\AMC@checkbox%
481 \AMC@use@box{\AMC@ovalbox@RF}%
482 \else%
483 \AMC@use@box{\AMC@ovalbox@R}%
484 \fi%
485 \fi%
486 }}%
487 }
488 \newcommand\AMC@shape@prepare@form{}
489 \newcommand\AMC@shape@form@base[5]{%
490 \ifx #2\AMC@checkbox%
491 \def\AMC@shape@form@ticked{true}%
492 \else%
493 \def\AMC@shape@form@ticked{false}%
494 \fi%
495 \AMC@tracebox{#3}{#4}{%
496 \CheckBox[checked=\AMC@shape@form@ticked,%
497 checkboxsymbol=\ding{110},name={#5},%
498 bordercolor=0 0 0,%
499 width=\AMC@boxedwidth,height=\AMC@boxedheight]{#5}}%
500 }%
501 }
502 \newcommand\AMC@shape@form[4]{%
503 \AMC@shape@form@base{#1}{#2}{#3}{#4}{\the\AMCid@etud:#4}%
504 }
505 \newcommand\AMC@shape@prepare@none{}
506 \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}}}`.

```

507 \def\AMCchoiceLabel#1{%
508 \ifAMC@inside@digit\arabic{#1}%

```

```

509 \else\Alph{#1}\fi%
510 }
511 \def\AMCchoiceLabelFormat#1{#1}
512 \newcounter{AMC@ncase}
513 \setcounter{AMC@ncase}{0}
514 \newcommand\AMC@answerBox[4]{%
515 \AMC@answerBox@{\ifx\@empty#1\@empty%
516 \AMCchoiceLabel{AMC@ncase}%
517 \else #1\fi}{#2}{#3}{#4}}

```

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

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

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

```

518 \newlength\AMC@boxedrule
519 \newlength\AMC@crossrule
520 \newlength\AMC@boxeddown
521 \newlength\AMC@boxedwidth
522 \newlength\AMC@boxedheight
523 \newlength\AMC@oval@radius
524 \newlength\AMC@outside@sep
525 \define@choicekey{AMCdim}{shape}{square,oval,form,none}{\def\AMC@shapename{#1}}
526 \define@key{AMCdim}{size}{\AMC@boxedwidth=#1\AMC@boxedheight=#1}
527 \define@key{AMCdim}{height}{\AMC@boxedheight=#1}
528 \define@key{AMCdim}{width}{\AMC@boxedwidth=#1}
529 \define@key{AMCdim}{rule}{\AMC@boxedrule=#1}
530 \define@key{AMCdim}{outsidesep}{\AMC@outside@sep=#1}
531 \define@key{AMCdim}{down}{\AMC@boxeddown=#1}
532 \define@key{AMCdim}{color}{\def\AMC@boxcolor{#1}}
533 \define@boolkey{AMCdim}{cross}{false}{}
534 \define@key{AMCdim}{crosschar}{\textbf{\textsf{X}}}{\def\AMC@crosschar{#1}}
535 \define@key{AMCdim}{crossrule}[1.5pt]{\AMC@crossrule=#1}
536 \def\AMC@shapeprepare{\csname AMC@shapeprepare@\AMC@shapename@ \endcsname}
537 \def\AMCboxStyle#1{%
538 \setkeys{AMCdim}{#1}%
539 \ifnum\AMC@boxedwidth<\AMC@boxedheight%
540 \AMC@oval@radius=\AMC@boxedwidth\divide\AMC@oval@radius\tw@%
541 \else%
542 \AMC@oval@radius=\AMC@boxedheight\divide\AMC@oval@radius\tw@%
543 \fi%
544 \AMC@shapeprepare%
545 }
546 \AMCboxStyle{shape=square,size=2.5ex,down=.4ex,rule=.5pt,outsidesep=.1em,color=black,cross,crosschar,crossrule}
547 \newcommand\AMCboxColor[1]{\AMCboxStyle{color=#1}}
548 \let\AMCboxDimensions=\AMCboxStyle

```

`\AMCboxOutsideLetter` Command `\AMC@box{<char>}{<answer>}` prints a box with character `<char>` inside, showing an-

```

\AMC@box
\AMC@formBox@
\AMC@formBox
outsideLabelFormat

```

swer $\langle answer \rangle$ (`\AMC@checkbox` to get a filled box), using global variables to identify the box (question and choice).

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

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

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

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

```

549 \def\AMCoutsideLabelFormat#1{#1}
550 \newcommand\AMCboxOutsideLetter[2]{#1\nobreak\hspace{.1em}\AMCoutsideLabelFormat{#2}}
551 \newif\ifAMC@printformoutside@%
552 \newcommand\ifAMC@printformoutside{%
553   \AMC@printformoutside@false%
554   \ifAMC@ensemble\ifAMC@outside@box%
555     \ifAMCformulaire@dedans\AMC@printformoutside@true\fi%
556     \ifAMC@zoneformulaire\AMC@printformoutside@true\fi%
557   \fi\fi%
558   \ifAMC@printformoutside@%
559 }
560 \newcommand\AMC@formBox@[4]{%
561   \ifAMC@printformoutside% letter to be written outside the box
562     \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
563   \else%
564     \AMC@answerBox@{#1}{#2}{#3}{#4}%
565   \fi%
566   \AMC@tracechar{#1}{#2}{#3}{#4}%
567 }
568 \newif\ifAMC@printkeyoutside@%
569 \newcommand\ifAMC@printkeyoutside{%
570   \AMC@printkeyoutside@false%
571   \ifAMC@ensemble%
572     \ifAMC@outside@box\AMC@printkeyoutside@true\fi%
573   \else%
574     \ifAMC@inside@box\else\AMC@printkeyoutside@true\fi%
575   \fi%
576   \ifAMC@printkeyoutside@%
577 }
578 \newcommand\AMC@keyBox@[4]{%
579   \ifAMC@printkeyoutside%
580     \AMCboxOutsideLetter{\AMC@answerBox@{#2}{#3}{#4}}{#1}%
581   \else%
582     \AMC@answerBox@{#1}{#2}{#3}{#4}%
583   \fi%
584   \AMC@tracechar{#1}{#2}{#3}{#4}%
585 }
586 \newcommand\AMC@formBox[4]{%
587   \AMC@formBox@{\ifx\@empty#1\@empty%
588     \AMCchoiceLabel\AMC@ncase}%
589   \else #1\fi}{#2}{#3}{#4}%
590 }
591 \newcommand{\AMC@box}[2]{%

```

```

592 \ifAMC@ensemble%
593 \ifAMC@zoneformulaire% for codes inside form sheet
594 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
595 \else%
596 \ifAMC@formulaire@dedans% for answer boxes inside form sheet
597 \protect\AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
598 \else% outside form sheet: not to be read during data capture
599 \AMC@formBox{#1}{#2}{1}{casequestion:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
600 \fi\fi%
601 \else% no separate sheet for answers: always read
602 \ifAMC@inside@box%
603 \AMC@formBox{#1}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
604 \else%
605 \AMC@formBox{}{#2}{1}{case:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}%
606 \fi%
607 \fi%
608 }

```

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.

```

609 \newif\ifAMCsz@logged\AMCsz@loggedfalse
610 \newcommand{\AMCscoreZone}[1]{%
611 \ifAMC@ensemble%
612 \ifAMC@formulaire@dedans%
613 \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
614 \else%
615 \AMC@tracebox{1}{scorequestion::\the\AMCid@quest,-1}{#1}%
616 \fi%
617 \else%
618 \AMC@tracebox{1}{score::\the\AMCid@quest,-1}{#1}%
619 \fi%
620 \ifAMCsz@logged\else%
621 \AMCmessage{VAR:scorezones=1}%
622 \global\AMCsz@loggedtrue%
623 \fi%
624 }

```

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`.

```

625 \def\AMCid@checkmax{60}
626 \def\AMC@NCBetud{12}
627 \def\AMC@NCBpage{6}
628 \def\AMC@NCBcheck{6}

```

```

629 \newlength{\AMC@CBtaille}\setlength{\AMC@CBtaille}{5cm}
630 \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 $\langle options \rangle$ 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:



```

631 \newtoks\AMCbin@sequence
632 \newcount\AMCbin@number
633 \newcount\AMCbin@digit
634 \newcount\AMCbin@id
635 \newcount\AMCbin@did
636 \newcount\AMCbin@endigits
637 \newdimen\AMCbin@hsep
638 \define@key{AMCbin}{ndigits}{\AMCbin@endigits=#1}
639 \define@key{AMCbin}{id}{\AMCbin@id=#1}
640 \define@key{AMCbin}{hsep}{\AMCbin@hsep=#1}
641 \define@key{AMCbin}{style}[]{\def\AMCbin@style{#1}}
642 \def\AMCbin@one{%
643   \ifnum\AMCbin@did>\z@%
644     \hspace{\AMCbin@hsep}%
645     \fi%
646     \advance\AMCbin@did\@ne%
647     \ifnum\AMCbin@id>0%
648       \AMC@answerBox@{}{\AMC@checkedbox}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
649       \else%
650       \AMC@answerBox@{}{\AMC@checkedbox}{1}{}%
651       \fi}
652 \def\AMCbin@zero{%
653   \ifnum\AMCbin@did>\z@%
654     \hspace{\AMCbin@hsep}%
655     \fi%
656     \advance\AMCbin@did\@ne%
657     \ifnum\AMCbin@id>0%
658       \AMC@answerBox@{}{1}{chiffre:\the\AMCbin@id,\the\AMCbin@did}%
659       \else%
660       \AMC@answerBox@{}{1}{}%
661       \fi}
662 \newcommand{\AMC@binaryCode}[2]{%
663 \setkeys{AMCbin}{ndigits=1,hsep=0pt,style}\setkeys{AMCbin}{#1}%
664 \AMCbin@did=\z@%

```

```

665 {\AMCboxDimensions{shape=square,size=.32cm,down=0pt,rule=.2pt,cross=false}\expandafter\AMCboxDimensions\
666 \AMCbin@digit=\z@%
667 \loop%
668 \ifnum\AMCbin@number>\z@%
669 \advance\AMCbin@digit\@ne%
670 \ifodd\AMCbin@number\AMCbin@sequence=\expandafter{\expandafter\AMCbin@one\the\AMCbin@sequence}%
671 \else\AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\fi%
672 \divide\AMCbin@number\tw@%
673 \repeat%
674 \loop\relax%
675 \ifnum\AMCbin@digit<\AMCbin@ndigits\advance\AMCbin@digit\@ne%
676 \AMCbin@sequence=\expandafter{\expandafter\AMCbin@zero\the\AMCbin@sequence}\repeat%
677 \the\AMCbin@sequence%
678 \ifnum\AMCbin@digit>\AMCbin@ndigits\PackageError{automultiplechoice}{Too low AMC@NCB value (got \the\AMC
679 }

```

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

```

680 \def\AMCbin@begin#1{\setkeys{AMCbin}{id=#1}}
681 \newcommand{\AMC@binaryBoxes}[2][1]{%
682 \AMC@binaryCode{ndigits=#1}{#2}%
683 }

```

4.9 Checking Environment

`\AMCcurrentenv` Sets the current environment as document.

```

684 \def\AMCcurrentenv{document}

```

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

```

685 \def\AMCif@env#1{
686   \def\AMC@tempenv{#1}%
687   \ifx\AMC@tempenv\AMCcurrentenv
688     \expandafter\@firstoftwo
689   \else
690     \expandafter\@secondoftwo
691   \fi
692 }

```

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.

```

693 \newcount\AMCtok@k
694 \newcount\AMCtok@max
695 \newcount\AMCtok@size

```

```

696 \newcommand{\nouveaugroupe}[2]{%
697   \expandafter\ifx\csname #1@k\endcsname\relax%
698     \expandafter\newcount\csname #1@k\endcsname%
699     \expandafter\newcount\csname AMC#1@j\endcsname%
700     \csname #1@k\endcsname=\z@\relax%
701     \csname AMC#1@j\endcsname=\z@\relax%
702     \setgroupmode{#1}{\AMCdefault@groupmode}%
703   \fi%
704 }
705 \newcommand\AMC@prepare@element[1]{%
706   \nouveaugroupe{#1}{}%
707   \global\advance\csname #1@k\endcsname\@ne\relax%
708   \AMCtok@k=\csname #1@k\endcsname%
709   \expandafter\ifx\csname #1@romannumeral\AMCtok@k\endcsname\relax%
710     \expandafter\newtoks\csname #1@romannumeral\AMCtok@k\endcsname\fi%
711 }
712 \newcommand{\element}[2]{%
713   \AMC@prepare@element{#1}%
714   \global\csname #1@romannumeral\AMCtok@k\endcsname={#2}%
715 }

```

`\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.

```

716 \def\AMCdefault@groupmode{fixed}
717 \newcommand{\setdefaultgroupmode}[1]{\def\AMCdefault@groupmode{#1}}
718 \newcommand{\setgroupmode}[2]{%
719   \expandafter\ifx\csname AMCgrouppre@#2\endcsname\relax%
720     \PackageError{automultiplechoice}{Unknown group mode for #1 : #2}%
721     {You asked to set group '#1' mode to '#2',
722      but '#2' is not a valid group mode}%
723   \else%
724     \expandafter\global\expandafter\def\csname AMC#1@mode\endcsname{#2}%
725   \fi%
726 }

```

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

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

```

727 \newcommand{\AMCgrouppre@fixed}[3]{%
728   \ifnum#3<\z@%
729     \csname AMC#1@j\endcsname=\z@%
730   \else%
731     \csname AMC#1@j\endcsname=#3%
732   \fi%
733 }

```

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.

```

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

```

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.

```

742 \newcount\AMC@imax
743 \newcommand{\AMCgrouppre@withoutreplacement}[3]{%
744   \ifnum#3<\z@%
745   \else%
746     \csname AMC#1@j\endcsname=#3%
747   \fi%
748   \ifnum\AMCtok@ik=\AMCloop@k%
749     \AMCtok@ik=\z@%
750   \fi%
751   \ifnum\AMCtok@ik=\z@%
752     \shufflegroup{#1}%
753   \else%
754     \AMC@imax=\AMCloop@k%
755     \advance\AMC@imax -#2\relax%
756     \ifnum\AMCtok@ik>\AMC@imax%
757       \shufflegroupslicel{#1}{\@ne}{\AMCtok@ik}%
758       \ifnum\AMCtok@ik<\AMCloop@k%
759         \advance\AMCtok@ik\@ne%
760         \shufflegroupslicel{#1}{\AMCtok@ik}{\AMCloop@k}%
761       \fi%
762     \fi%
763   \fi%
764 }

```

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

```

765 \newcommand{\AMCgrouppre@cyclic}[3]{%
766   \ifnum#3<\z@%
767   \else%
768     \csname AMC#1@j\endcsname=#3%
769   \fi%
770 }

```

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

```
771 \newcommand{\AMCgroup@pre}[4]{%
772   \csname AMCgroup@pre@#1\endcsname{#2}{#3}{#4}%
773 }
```

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

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

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

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

```
774 \newcommand{\shufflegroup}[1]{%
775   \ifAMC@shuffleG{\AMC@shuffletoks{\number\csname #1@k\endcsname}{#1@}}\fi%
776 }
777 \newcommand{\shufflegroupslices}[3]{%
778   \ifAMC@shuffleG{\AMC@shuffletoks[#2]{#3}{#1@}}\fi%
779 }
780 \newcount\AMCtok@ik
781 \newcount\AMCloop@k
782 \newcommand{\AMCgrouploop@prep}[3]{%
783   \AMCtok@size=#1\relax%
784   \ifAMC@fullGroups\AMCtok@size=\m@ne\fi%
785   \ifnum\AMCtok@size<\z@%
786     \AMCtok@size=\csname #2@k\endcsname%
787   \fi%
788   \AMCtok@ik=\csname AMC#2@j\endcsname%
789   \AMCloop@k=\csname #2@k\endcsname%
790   \expandafter\ifx\csname AMC#2@mode\endcsname\relax%
791     \PackageError{automultiplechoice}{No group mode for #2}%
792     {No mode has been defined for group '#2'. This should not occur...}%
793   \fi%
794   \AMCgroup@pre{\csname AMC#2@mode\endcsname}{#2}{\the\AMCtok@size}{#3}%
795 }
796 \newcommand{\AMCgrouploop@next}[1]{%
797   \global\advance\csname AMC#1@j\endcsname\@ne\relax%
798   \expandafter\ifnum\csname AMC#1@j\endcsname>\AMCloop@k\relax%
799     \global\csname AMC#1@j\endcsname=\@ne%
800   \fi%
801   \AMCtok@ik=\csname AMC#1@j\endcsname%
802   \advance\AMCtok@size\m@ne%
803 }
804 \newcommand{\insertgroupfrom}[3][-1]{%
805   \ifnum#1=0%
806     \else%
807       \AMCgrouploop@prep{#1}{#2}{#3}%
808       {\loop%
809         \AMCgrouploop@next{#2}%
810         {\the\csname #2@romannumeral\AMCtok@ik\endcsname}%
811         \ifnum\AMCtok@size>\z@\repeat}%
812       \fi%
813 }
```

```

814 \newcommand{\insertgroup}[2][-1]{%
815   \insertgroupfrom[#1]{#2}{-1}%
816 }

```

`\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.

```

817 \newcommand{\cleargroup}[1]{%
818   \nouveau groupe{#1}{}%
819   \csname #1@k\endcsname=\z@\relax%
820   \csname AMC#1@j\endcsname=\z@\relax%
821 }
822 \newcommand{\copygroupfrom}[4][-1]{%
823   \ifnum#1=0%
824     \else%
825       \AMCgrouploop@prep{#1}{#2}{#4}%
826       {\loop%
827         \AMCgrouploop@next{#2}%
828         \AMC@prepare@element{#3}%
829         \global\csname #3@romannumeral\AMCtok@k\endcsname=\csname #2@romannumeral\AMCtok@ik\endcsname%
830         \ifnum\AMCtok@size>\z@\repeat}%
831       \fi%
832 }
833 \newcommand{\copygroup}[3][-1]{%
834   \copygroupfrom[#1]{#2}{#3}{-1}%
835 }

```

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`.

```

836 \newcount\AMCrep@count
837 \newcount\AMCrep@nn
838 \newcount\AMCrep@nnmax
839 \AMCload@counter=199
840 \@whilenum\AMCload@counter>0\do{%
841   \expandafter\newtoks\csname reponse@\romannumeral\AMCload@counter\endcsname%
842   \advance\AMCload@counter\m@ne%
843 }

```

`\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>`.

```
844 \newcommand\AMCload@reponse[2]{%
845   \global\advance\AMCload@counter\@ne\relax%
846   \global\csname reponse@\romannumeral\AMCload@counter\endcsname%
847   =\expandafter{\expandafter\AMCrep@count\expandafter=#2 #1}%
848 }
849 \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.

```
850 \def\shuffle@it{\AMC@shuffletoks{\number\AMCload@counter}{reponse@}}
851 \newcount\AMCnum@questions
852 \newcommand\AMCdump@reponses{%
853   \global\AMCnum@questions=\AMCload@counter%
854   \@whilenum\AMCload@counter>0\do{%
855     \the\csname reponse@\romannumeral\AMCload@counter\endcsname%
856     \advance\AMCload@counter\m@ne}}
```

4.11.1 Managing answers

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

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

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

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

```
857 \newcommand\AMCrep@init[1]{%
858   \ifAMC@ordre\AMCrep@o\else%
859     \csname AMCrep@#1\endcsname\fi\AMCload@counter=\z@}
860 \newcommand\AMCrep@o{%
861   \def\AMCload@@reponse{\AMCrien@deux}\def\AMCrep@fini{}}
862 \newcommand\AMCrep@r{%
863   \def\AMCload@@reponse{\AMCload@reponse}%
864   \def\AMCrep@fini{\shuffle@it\AMCdump@reponses}}
865 \newcount\AMCrep@count
866 \newcommand\lastchoices{%
867   \AMCrep@count=\AMCrep@count%
868   \AMCrep@fini\AMCrep@init{o}%}
```

```

869 \AMCrep@count=\AMCrep@count}
870 \newcommand\@aucune{\emph{\AMC@loc@none}}
871 \newcommand\AMC@fin@rep{%
872 \ifAMCcomplete@multi\ifAMCtype@multi%
873 \lastchoices\AMCrep@count=-1%
874 \ifAMCune@bonne\wrongchoice{\@aucune}\else%
875 \ifAMC@postcorrect\wrongchoice{\@aucune}\else\correctchoice{\@aucune}\fi%
876 \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.

```

877 \def\AMCformBeforeQuestion{\vspace{\AMCformVSpace}\par}
878 \def\AMCformAfterQuestion{\ifAMC@asqbloc\egroup\fi}
879 \def\AMCformQuestion#1{\AMC@loc@qf{#1}}
880 \def\AMCformQuestionN{\AMCformQuestion{\AMC@qaff}}
881 \def\AMCformQuestionA{%
882 \setcounter{AMC@ncase}{0}%
883 \AMCformBeforeQuestion%
884 \ifAMC@asqbloc\vbox\bgroup\fi%
885 \ifx\@empty\AMC@sza@callout\@empty\else%
886 \csname\AMC@sza@callout\endcsname%
887 \fi%
888 \AMCformQuestionN%
889 \ifx\@empty\AMC@sza@callin\@empty\else%
890 \csname\AMC@sza@callin\endcsname%
891 \fi%
892 }
893 \def\AMCformAnswer#1{\hspace{\AMCformHSpace} #1}
894 \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.

```

895 \ExplSyntaxOn
896
897 \prg_set_conditional:Nnn \amc_if_separate_question: { p , T } {
898 \ifAMC@ensemble

```

```

899     \ifAMC@zoneformulaire
900         \prg_return_false:
901     \else
902         \prg_return_true:
903     \fi
904 \else
905     \prg_return_false:
906 \fi
907 }
908 \cs_new_eq:NN \AMC@if@separate@question \amc_if_separate_question:T
909
910 \int_new:N \amc_memory_elts_count
911
912 \cs_new:Nn \amc_clear_memory: { \int_gzero:N \amc_memory_elts_count }
913 \cs_new_eq:NN \AMC@mem@clear \amc_clear_memory:
914
915 \cs_new:Npn \amc_memory_elt_i:n #1 {
916     amc_memory_elts_ \int_to_alph:n { #1 }
917 }
918 \cs_new:Nn \amc_memory_current_elt: {
919     \amc_memory_elt_i:n \amc_memory_elts_count
920 }
921 \cs_new:Npn \amc_memory_vars_i:n #1 {
922     amc_memory_vars_ \int_to_alph:n { #1 }
923 }
924 \cs_new:Nn \amc_memory_current_vars: {
925     \amc_memory_vars_i:n \amc_memory_elts_count
926 }
927
928 \cs_new:Nn \amc_add_memory_elt: {
929     \int_gincr:N \amc_memory_elts_count
930     \tl_gclear_new:c { \amc_memory_current_elt: }
931     \tl_gclear_new:c { \amc_memory_current_vars: }
932 }
933 \cs_new_eq:NN \AMC@mem@next \amc_add_memory_elt:
934
935 \cs_new:Npn \amc_add_to_memory:n #1 {
936     \tl_gput_right:cn { \amc_memory_current_elt: } { #1 }
937 }
938 \cs_new_eq:NN \AMC@mem@add \amc_add_to_memory:n
939
940 \cs_new:Npn \amc_add_to_vars:n #1 {
941     \tl_gput_right:cn { \amc_memory_current_vars: } { #1 }
942 }
943 \cs_new_eq:NN \AMC@mem@addvar \amc_add_to_vars:n
944
945 \cs_new:Npn \amc_add_qidaffname:nnn #1#2#3 {
946     \amc_add_to_vars:n {\AMCid@quest=#1\setcounter{AMCquestionaff}{#2}%
947         \global\def\AMCid@name{#3}}
948 }
949 \cs_generate_variant:Nn \amc_add_qidaffname:nnn { xxx }
950 \cs_new_eq:NN \AMC@mem@qidaffname \amc_add_qidaffname:xxx
951

```

```

952 \cs_new:Npn \amc_mem_elt_cat:n #1 {
953   \amc_add_to_vars:n { \def\AMCmem@elt@cat{ #1 } }
954 }
955 \cs_generate_variant:Nn \amc_mem_elt_cat:n { x }
956 \cs_new_eq:NN \AMC@mem@category \amc_mem_elt_cat:x
957
958 \cs_new:Npn \amc_add_aid:n #1 {
959   \amc_add_to_memory:n {\AMCrep@count=#1}
960 }
961 \cs_generate_variant:Nn \amc_add_aid:n { x }
962 \cs_new_eq:NN \AMC@mem@aid \amc_add_aid:x
963
964 \cs_new:Npn \amc_if_category_is_p:n #1 {
965   \str_if_eq_p:on { \AMCmem@elt@cat } { #1 }
966 }
967 \cs_new:Npn \amc_use_memory:n #1 {
968   \int_step_inline:nnnn { 1 } { 1 } \amc_memory_elts_count {
969     \def\AMCmem@elt@cat{ plain }
970     \tl_use:c { \amc_memory_vars_i:n { ##1 } }
971     \bool_if:nTF { #1 } {
972       \tl_use:c { \amc_memory_elt_i:n { ##1 } }
973     } { }
974   }
975 }
976 \cs_new:Nn \amc_use_memory: { \amc_use_memory:n { \c_true_bool } }
977 \cs_new_eq:NN \AMC@mem@show \amc_use_memory:
978 \cs_new_eq:NN \AMC@mem@show@filter \amc_use_memory:n
979 \cs_new_eq:NN \AMC@if@category \amc_if_category_is_p:n
980
981 \ExplSyntaxOff
982 \newcommand\AMC@mem@add@ifneeded[1]{%
983   \AMC@if@separate@question{%
984     \AMC@mem@add{#1}%
985   }%
986 }
987 \newcommand\AMC@mem@addsingle@ifneeded[2]{%
988   \AMC@if@separate@question{%
989     \AMC@mem@next%
990     \AMC@mem@category{#2}%
991     \AMC@mem@add{#1}%
992   }%
993 }
994 \newcommand\AMC@mem@answer[1]{%
995   \addtocounter{AMC@ncase}{1}%
996   \AMC@if@separate@question{%
997     \AMC@mem@aid{\the\AMCrep@count}%
998     \AMC@mem@add{\AMCformAnswerA{#1}}%
999   }%
1000 }
1001 \newcommand\AMC@mem@openQuestion{%
1002   \AMC@if@separate@question{%
1003     \AMC@mem@next%
1004     \AMC@mem@qidaffname{\the\AMCid@quest}{\arabic{AMCquestionaff}}{\AMCid@name}%

```

```

1005 \AMC@mem@add{\AMCformQuestionA}%
1006 }%
1007 }
1008 \def\AMCformBegin{%
1009 \AMC@zoneformulairetrue\setcounter{section}{0}%
1010 \ifAMC@ensemble\ifAMC@automarks\pagestyle{AMCpageFull}\fi\fi%
1011 }
1012 \newcommand\AMCform{%
1013 \ifAMC@ensemble\AMCformulaire@dedanstrue%
1014 \AMC@mem@show%
1015 \fi}
1016 \newcommand\AMCformFilter[1]{%
1017 \ifAMC@ensemble\AMCformulaire@dedanstrue%
1018 \AMC@mem@show@filter{#1}%
1019 \fi}
1020 \newif\ifAMC@keepmemory
1021 \newcommand\AMCformS{%
1022 \ifAMC@ensemble\AMCformulaire@dedanstrue%
1023 \AMCmessage{BR=0}\AMC@mem@show%
1024 \global\AMC@keepmemorytrue%
1025 \fi}

```

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

The argument *<options>* is a key-value list, where:

- `nanswers=<n>` gives the number of answers that are labeled in the table (defaults to the maximal number of answers of the subject).
- `ncols=<n>` gives the number of columns to use (defaults to the largest possible number of columns).
- `idtext=<text>` gives a text to be inserted before each question number.
- `idtitle=<text>` 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).

```

1026 \newcounter{AMC@specsi}
1027 \def\AMC@formspecs#1{%
1028 \noindent\hbox to \AMCformTH{\hspace*{\fill}\textbf{\AMC@tf{idtitle}}\hspace{\AMCformHSpace}}%
1029 \setcounter{AMC@specsi}{0}%
1030 \loop%
1031 \addtocounter{AMC@specsi}{1}%
1032 \AMCformAnswer{\bf\AMCchoiceLabel{AMC@specsi}}%
1033 \ifnum\value{AMC@specsi}<#1\repeat%
1034 }
1035 \newcount\AMC@coli
1036 \def\AMC@formcolspecs#1#2{%
1037 \columnseprule=0pt%

```

```

1038 \begin{multicols}{#1}
1039 \AMC@coli=\z@%
1040 \@whilenum\AMC@coli<#1\do{
1041   \advance\AMC@coli\@ne%
1042   \AMC@formspecs{#2}\par
1043 }
1044 \end{multicols}
1045 }}
1046 \newcount\AMC@tf@ncols
1047 \newdimen\AMCformTH
1048 \newdimen\AMC@tfcolw
1049 \newdimen\AMC@tfaw
1050 \def\AMC@tf@idtext{}
1051 \def\AMC@tf@idtitle{}
1052 \newbox\AMC@tfbox
1053 \define@key{AMCtf}{nanswers}[0]{\def\AMC@tf@nanswers{#1}}
1054 \define@key{AMCtf}{ncols}[0]{\AMC@tf@ncols=#1}
1055 \define@key{AMCtf}{idtext}{\def\AMC@tf@idtext{#1}}
1056 \define@key{AMCtf}{idtitle}{\def\AMC@tf@idtitle{#1}}
1057 \define@key{AMCtf}{idwidth}[Opt]{\AMCformTH=#1}
1058 \define@key{AMCtf}{columnsep}[0.5em]{\columnsep=#1}
1059 \define@key{AMCtf}{columnseprule}[0.5pt]{\columnseprule=#1}
1060 \define@boolkey{AMCtf}{headers}[true]{}
1061 \define@boolkey{AMCtf}{inside}[false]{}
1062 \setkeys{AMCtf}{nanswers,ncols,idwidth,headers,inside}
1063 \newcommand\AMCtableForm[1][{}]{%
1064   \setkeys{AMCtf}{columnsep,columnseprule}
1065   \setkeys{AMCtf}{#1}%
1066   \ifnum\AMC@tf@nanswers=0%
1067     \def\AMC@tf@nanswers{\the\AMCrep@nnmax}%
1068   \fi%
1069   \newdimen\AMC@tf@colW\AMC@tf@colW=\AMCformHSpace%
1070   \advance\AMC@tf@colW by \AMC@boxedwidth%
1071   \def\AMCformBeforeQuestion{\vspace{\AMCformVSpace}\par}%
1072   \def\AMCformQuestion##1{\noindent\hbox to \AMCformTH{\hspace*{\fill}\textbf{\AMC@tf@idtext{##1}}\hspace*{\fill}}}
1073   \def\AMCformAnswer##1{\hbox to \AMC@tf@colW{\hspace*{\fill}##1\hspace*{\fill}}}%
1074   \ifnum\AMCformTH=0%
1075     \setbox\AMC@tfbox=\hbox{\textbf{\AMC@tf@idtext{999}}}%
1076     \AMCformTH=\wd\AMC@tfbox%
1077   \fi%
1078   \ifnum\AMC@tf@ncols=0%
1079     \AMC@tfcolw=\AMC@tf@nanswers\AMC@tf@colW%
1080     \advance\AMC@tfcolw by \AMCformTH%
1081     \advance\AMC@tfcolw by \columnsep%
1082     \AMC@tfaw=\linewidth%
1083     \loop%
1084       \ifnum\AMC@tfaw>\AMC@tfcolw%
1085         \advance\AMC@tfaw by -\AMC@tfcolw%
1086         \advance\AMC@tf@ncols\@ne%
1087       \repeat%
1088   \fi%
1089   \ifKV@AMCtf@headers%
1090     \AMC@formcolspecs{\the\AMC@tf@ncols}{\AMC@tf@nanswers}%

```

```

1091 \vspace{\AMCformVSpace}\vspace{-2\multicolsep}
1092 \fi%
1093 {\raggedcolumns
1094 \begin{multicols}{\the\AMC@tf@ncols}
1095 \ifKV@AMCtf@inside\else\def\AMCchoiceLabel##1{}\fi%
1096 \AMCform
1097 \end{multicols}}
1098 }

```

`\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.

```

1099 \newcommand{\AMCsectionNumbered}[1]{%
1100 \section{#1}\AMC@mem@addsingle@ifneeded{\section{#1}}{section}}
1101 \newcommand{\AMCsubsectionNumbered}[1]{%
1102 \subsection{#1}\AMC@mem@addsingle@ifneeded{\subsection{#1}}{subsection}}
1103 \newcommand{\AMCsectionStar}[1]{%
1104 \section*{#1}\AMC@mem@addsingle@ifneeded{\section*{#1}}{section}}
1105 \newcommand{\AMCsubsectionStar}[1]{%
1106 \subsection*{#1}\AMC@mem@addsingle@ifneeded{\subsection*{#1}}{subsection}}
1107 \def\AMCsection{\@ifstar\AMCsectionStar\AMCsectionNumbered}
1108 \def\AMCsubsection{\@ifstar\AMCsubsectionStar\AMCsubsectionNumbered}

```

4.11.3 Formatting answers

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

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

```

tikz-multi (env.) 1109 \def\AMCBoxedAnswers{\AMC@rbloctrue}
\AMCBoxedAnswers 1110 \newcommand{\start@Answers}{%
1111 \global\AMCrep@count=\z@%
1112 \global\AMCrep@nn=\z@%
1113 }
1114 \newenvironment{choices}[1][r]{%
1115 \def\une@rep{\AMCrep@itemize}%
1116 \ifAMC@rbloc\def\une@rep{\AMCrep@bloc}%
1117 \else\begin{itemize}\setlength{\itemsep}{\AMCinterIrep}\fi%
1118 \AMCrep@init{#1}}%
1119 {\AMC@fin@rep\ifAMC@rbloc\else\end{itemize}\fi}
1120 \newenvironment{choiceshoriz}[1][r]{%
1121 \def\une@rep{\AMCrep@ligne}\AMCrep@init{#1}%
1122 \par\begin{center}}%
1123 {\AMC@fin@rep\end{center}}
1124 \newenvironment{choicescustom}[1][r]{%
1125 \def\une@rep{\AMCrep@perso}\AMCrep@init{#1}%
1126 \AMCbeginAnswer\ignorespaces}%
1127 {\AMC@fin@rep\AMCendAnswer}
1128 \newenvironment{tikz-single}[1][r]{
1129 \def\une@rep{\AMCrep@tikz}\AMCrep@init{#1}%
1130 \begin{tikzpicture}}{\AMC@fin@rep\end{tikzpicture}}
1131 \newenvironment{tikz-multi}[1][r]{
1132 \def\une@rep{\AMCrep@tikzmat}\AMCrep@init{#1}%
1133 \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` 1134 `\newcommand\AMCrep@bloc[2]{\AMC@mem@answer{#1}%`
`\AMCrep@ligne` 1135 `\par%`
`\AMCrep@perso` 1136 `\ifAMC@textPos\ vbox\bgroup\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\hbox%`
1137 `\noindent\begin{minipage}{\linewidth}%`
1138 `\begin{itemize}\item{#1} #2\end{itemize}\end{minipage}%`
1139 `\ifAMC@textPos\AMC@tracepos{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}\egroup\AMC@tra`
1140 `\vspace{\AMCinterBrep}}`
1141 `\newcommand\AMCrep@tikz[5]{\AMC@mem@answer{#1}\node[#4] (lab\thecsvrow) at (#3) {#2} node[#5] (box\the`
1142 `\newcommand\AMCrep@tikzmat[5]{\AMC@mem@answer{#1}\node[#5] (box\thecsvrow) at (#3) {#1} node[#4] (lab\th`
1143 `\newcommand\AMCrep@itemize[2]{\AMC@mem@answer{#1}\item{#1} #2}`
1144 `\newlength\AMChorizAnswerSep`
1145 `\setlength{\AMChorizAnswerSep}{3em plus 4em}`
1146 `\newlength\AMChorizBoxSep`
1147 `\setlength{\AMChorizBoxSep}{1em}`
1148 `\newcommand\AMCrep@ligne[2]{\AMC@mem@answer{#1}%`
1149 `\ifAMC@textPos%`
1150 `\mbox{\AMC@tracebox{1}{atext:\AMCid@name:\the\AMCid@quest,\the\AMCrep@count}{#1}\hspace*{\AMChorizBoxS`
1151 `\else%`
1152 `\mbox{#1}\hspace*{\AMChorizBoxSep}#2}%`
1153 `\fi\hspace{\AMChorizAnswerSep}}`
1154 `\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.
1155 `\def\AMCbeginAnswer{}`
1156 `\def\AMCanswer#1#2{#1 #2}`
1157 `\def\AMCendAnswer{}`

`\answer` The commands `\correctchoice` and `\wrongchoice` are used inside `choices`-like environments
`\correctchoice` to give the proposed answers and specify if they are to be ticked by the students or not.
`\wrongchoice` 1158 `\newcommand{\next@Answer}{%`
1159 `\global\advance\AMCrep@count\@ne\relax%`
1160 `\global\advance\AMCrep@nn\@ne\relax%`
1161 `\ifnum\AMCrep@nn>\AMCrep@nnmax%`
1162 `\global\AMCrep@nnmax=\AMCrep@nn%`
1163 `\fi%`
1164 `}`
1165 `\newcommand{\correctchoice}[2] []{%`
1166 `\next@Answer%`
1167 `\ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%`
1168 `\global\AMCune@bonnettrue%`
1169 `\AMCload@@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkedbox}}%`
1170 `\else\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}\ignorespaces}`
1171 `\newcommand{\wrongchoice}[2] []{%`
1172 `\next@Answer%`
1173 `\ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:M}\fi%`
1174 `\AMCload@@reponse{\une@rep{\AMC@box{#1}{\fi}{#2}}{\the\AMCrep@count}}%`
1175 `\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](http://www.sgmoye.com) comments on tex.stackexchange.com) prints a marginal note in the left or right margin, depending on current the position (usefull in `multicols` environment).

```

1176 \newcommand{\AMCemptybox}[3]{%
1177   \sbox0{}\wd0=#1\ht0=#2\dp0=#3\relax\box0}%
1178 \newlength\AMC@mn@test
1179 \newlength\AMC@mn@sep\AMC@mn@sep=4mm
1180 \newlength\AMC@mn@leftmargin
1181 \newlength\AMC@mn@rightmargin
1182 \newcommand\AMCmarginNote[1]{%
1183   \begin{tikzpicture}[remember picture,overlay]%
1184     \coordinate (here) at (0,0);%
1185     \pgfextractx{\AMC@mn@test}{\pgfpointdiff{\pgfpointorigin}%
1186       {\pgfpointanchor{current page}{center}}}%
1187     \ifodd\thepage%
1188       \AMC@mn@leftmargin=\oddsidemargin%
1189       \AMC@mn@rightmargin=\evensidemargin%
1190     \else
1191       \AMC@mn@leftmargin=\evensidemargin%
1192       \AMC@mn@rightmargin=\oddsidemargin%
1193     \fi
1194     \ifdim\AMC@mn@test < 1cm%
1195       \draw (current page.east |- here)+(-\AMC@mn@rightmargin-1in+\AMC@mn@sep,0pt) node[anchor=text,align=right]{#1}%
1196     \else%
1197       \draw (current page.west |- here)+(0cm,0pt) node[anchor=text,align=right,text width=\AMC@mn@leftmargin]{#1}%
1198     \fi%
1199   \end{tikzpicture}%
1200 }
```

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)

```

1201 \newcommand{\AMC@sz@box}{\AMCemptybox{\AMC@sz@width}{\AMC@sz@height}{\AMC@sz@depth}}
1202 %
1203 \newcommand{\AMC@sz@callin@question}{\AMCscoreZone{\AMC@sz@box}}
1204 %
1205 \newcommand{\AMC@sz@callout@margin}{\hspace{0pt}\marginpar{\AMCscoreZone{\AMC@sz@box}}}
1206 %
1207 \newcommand{\AMC@sz@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper margins}}
1208 \newcommand{\AMC@sz@callout@margins}{\hspace{0pt}\AMCmarginNote{\AMCscoreZone{\AMC@sz@box}}}
```

Let us now set up options handling.

```

1209 \newlength\AMC@sz@width
1210 \newlength\AMC@sz@height
1211 \newlength\AMC@sz@depth
1212 \def\AMC@sz@callout{}
1213 \def\AMC@sz@callin{}
1214 \define@key{AMC@sz}{width}{\AMC@sz@width=#1}
1215 \define@key{AMC@sz}{height}{\AMC@sz@height=#1}
1216 \define@key{AMC@sz}{depth}{\AMC@sz@depth=#1}
1217 \define@key{AMC@sz}{calloutside}{\def\AMC@sz@callout{#1}}
1218 \define@key{AMC@sz}{callinside}{\def\AMC@sz@callin{#1}}
1219 \define@choicekey{AMC@sz}{position}{none,question,margin,margins}{%
1220   \ifcsname AMC@sz@callout@#1\endcsname%
1221     \def\AMC@sz@callout{AMC@sz@callout@#1}%
1222   \else%
1223     \def\AMC@sz@callout{}%
1224   \fi%
1225   \ifcsname AMC@sz@callin@#1\endcsname%
1226     \def\AMC@sz@callin{AMC@sz@callin@#1}%
1227   \else%
1228     \def\AMC@sz@callin{}%
1229   \fi%
1230   \ifcsname AMC@sz@init@#1\endcsname%
1231     \csname AMC@sz@init@#1\endcsname%
1232   \fi%
1233 }
1234 \newcommand{\AMCsetScoreZone}[1]{\setkeys{AMC@sz}{#1}}
1235 \AMCsetScoreZone{width=1.5em,height=1.5ex,depth=.5ex,position=none}

```

And do the same for \AMCsetScoreZoneAnswerSheet...

```

1236 \newcommand{\AMC@sza@box}{\AMCemptybox{\AMC@sza@width}{\AMC@sza@height}{\AMC@sza@depth}}
1237 %
1238 \newcommand{\AMC@sza@init@none}{}
1239 \newcommand{\AMC@sza@callout@none}{}
1240 \newcommand{\AMC@sza@callin@none}{}
1241 %
1242 \newcommand{\AMC@sza@init@question}{}
1243 \newcommand{\AMC@sza@callout@question}{}
1244 \newcommand{\AMC@sza@callin@question}{\AMCscoreZone{\AMC@sza@box}}
1245 %
1246 \newcommand{\AMC@sza@init@margin}{}
1247 \newcommand{\AMC@sza@callout@margin}{\hspace{Opt}\marginpar{\AMCscoreZone{\AMC@sza@box}}}
1248 \newcommand{\AMC@sza@callin@margin}{}
1249 %
1250 \newcommand{\AMC@sza@init@margins}{\PackageWarning{automultiplechoice}{Please run twice to get proper ma
1251 \newcommand{\AMC@sza@callout@margins}{\hspace{Opt}\AMCmarginNote{\AMCscoreZone{\AMC@sza@box}}}
1252 \newcommand{\AMC@sza@callin@margins}{}
1253 %
1254 \newlength\AMC@sza@width
1255 \newlength\AMC@sza@height
1256 \newlength\AMC@sza@depth
1257 \def\AMC@sza@callout{}
1258 \def\AMC@sza@callin{}

```

```

1259 \define@key{AMCsza}{width}{\AMC@sza@width=#1}
1260 \define@key{AMCsza}{height}{\AMC@sza@height=#1}
1261 \define@key{AMCsza}{depth}{\AMC@sza@depth=#1}
1262 \define@key{AMCsza}{calloutsid}{\def\AMC@sza@callout{#1}}
1263 \define@key{AMCsza}{callinsid}{\def\AMC@sza@callin{#1}}
1264 \define@choicekey{AMCsza}{position}{none,question,margin,margins}{%
1265   \ifcsname AMC@sza@callout@#1\endcsname%
1266     \def\AMC@sza@callout{AMC@sza@callout@#1}%
1267   \else%
1268     \def\AMC@sza@callout{}%
1269   \fi%
1270   \ifcsname AMC@sza@callin@#1\endcsname%
1271     \def\AMC@sza@callin{AMC@sza@callin@#1}%
1272   \else%
1273     \def\AMC@sza@callin{}%
1274   \fi%
1275   \ifcsname AMC@sza@init@#1\endcsname%
1276     \csname AMC@sza@init@#1\endcsname%
1277   \fi%
1278 }
1279 \newcommand{\AMCsetScoreZoneAnswerSheet}[1]{\setkeys{AMCsza}{#1}}
1280 \AMCsetScoreZoneAnswerSheet{width=1.5em,height=1.5ex,depth=.5ex,position=none}
1281 \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.

```

1282 \newcounter{AMCquestionaff}
1283 \newcommand{\AMCnumero}[1]{\setcounter{AMCquestionaff}{#1}\addtocounter{AMCquestionaff}{-1}}
1284 \AtBeginDocument{%
1285   \ifx\@skiphyperreftrue\@undefined%
1286     \expandafter\newif\csname if@skiphyperref\endcsname%
1287   \fi%
1288 }
1289 \newcommand\AMC@stepQuestion{\ifAMCquestionNumber\@skiphyperreftrue\refstepcounter{AMCquestionaff}\@skip
1290 \newcommand\AMC@qaff{\arabic{AMCquestionaff}}

```

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

```

1291 \def\AMCbeforeQuestion{\ifAMC@qbloc\else\par\noindent\fi}
1292 \def\AMCbeginQuestion#1#2{\noindent\AMC@loc@q{#1}{#2}%
1293   \ifx\@empty\AMC@sz@callin\@empty\hspace*{1em}\fi%
1294 }
1295 \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.

`questionmult (env.)`

`variable-single (env.)`

`variable-multi (env.)`

`questionouverte (env.)`

`\ouverte@vs`

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

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

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

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

```

1296 \ifx\question\undefined\else\let\question\undefined\fi
1297 \def\AMCnobloc{\AMC@qblocfalse}
1298 \def\AMCbloc{\AMC@qbloctrue}
1299 \newcommand\AMCstartWithQuestion[1]{%
1300   \global\def\AMCid@name{#1}\AMC@affecte{#1}{\AMCid@quest}%
1301   \ifAMC@calibration%
1302     \AMCmessage{Q=\the\AMCid@quest}%
1303     \immediate\write\AMC@XYFILE{\string\question{\the\AMCid@quest}{\AMCid@name}}%
1304   \fi%
1305   \start@Answers%
1306 }
1307 \newcommand\AMCexternalQuestion[2]{%
1308   \AMCstartWithQuestion{#1}%
1309   \ifAMC@calibration%
1310     \AMCmessage{B=MAX=#2}%
1311     \AMCmessage{MULT}%
1312     \AMCmessage{FQ}%
1313   \fi%
1314 }
1315 \newenvironment{question}[2][{}]{%
1316   \def\AMCcurrentenv{question}%
1317   \AMC@stepQuestion%
1318   \AMCstartWithQuestion{#2}%
1319   \AMCbeforeQuestion%
1320   \ifx\@empty\AMC@sz@callout\@empty\else%
1321     \csname\AMC@sz@callout\endcsname%
1322   \fi%
1323   \AMCtype@multifalse\ifAMC@qbloc\ifAMC@textPos\ vbox\bgroup\AMC@tracemos{1}{qtext:#2:\the\AMCid@quest,0}
1324   \ifAMC@affichekeys\index{\texttt{#2}}\ifAMC@keyline[\texttt{#2}]\newline\fi\fi%
1325   \AMCbeginQuestion{\ifAMC@affichekeys\ifAMC@ensemble\AMC@qaff\ \fi\ifAMC@keyline\else[\texttt{#2}]\fi}
1326   \ifAMC@calibration\immediate\write\AMC@XYFILE{\string\questionnum{\the\AMCid@etud}{\the\AMCid@quest}{\the\AMCid@score}}
1327   \ifx\@empty\AMC@sz@callin\@empty\else%
1328     \csname\AMC@sz@callin\endcsname%
1329   \fi%
1330   \AMCformulaire@dedansfalse\setcounter{AMC@ncase}{0}%
1331   \AMC@mem@openQuestion}%
1332 {\ifAMC@qbloc\end{minipage}\ifAMC@textPos\AMC@tracemos{1}{qtext:\AMCid@name:\the\AMCid@quest,0}\egroup\AMC@endQuestion}
1333 \newenvironment{questionmult}[1]{%
1334   \AMC@un@bonnefalse\begin{question}[\{\multiSymbole\}]{#1}%
1335   \AMCtype@multitrue\ifAMC@calibration%
1336     \AMCmessage{MULT}\fi%
1337 }{\end{question}}
1338 \newif\ifSurveySingleAnswer\SurveySingleAnswerfalse

```

```

1339 \newenvironment{variable-single}[2]
1340 {\def\AMCbeginQuestion##1##2{}
1341 \begin{question}{#1}\scoring{v=#2}\QuestionIndicative\ifSurveySingleAnswer\else\AMCmessage{MULT}\fi
1342 \begin{tikz-single}[o]}
1343 {\end{tikz-single}
1344 \end{question}}
1345 \newenvironment{variable-multi}[4]
1346 {\def\AMCbeginQuestion##1##2{}
1347 \begin{question}{#1}\scoring{v=#4}\QuestionIndicative\ifSurveySingleAnswer\else\AMCmessage{MULT}\fi
1348 \begin{tikz-multi}[o]
1349 \node[#3] (var) at (0,0) {#2};}
1350 {\end{tikz-multi}
1351 \end{question}}
1352 \newenvironment{questionmultx}[1]{%
1353   \begingroup\def\multiSymbole{}\begin{questionmult}{#1}}%
1354 {\end{questionmult}\endgroup}
1355 \newdimen\ouverte@vs
1356 \newenvironment{questionouverte}[1][3cm]{%
1357   \AMC@stepQuestion%
1358   \AMCtype@multifalse\ouverte@vs=#1%
1359   \ifAMC@qbloc\noindent\begin{minipage}{\linewidth}\fi%
1360   \AMCbeginQuestion{\AMC@qaff}{}}%
1361 {\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.

```

1362 \newcommand{\explain}[1]{%
1363 \ifAMC@correc%
1364   \AMCif@env{question}{\par\noindent\AMC@loc@explain #1}{\AMC@error@explain}\vspace{1ex}%
1365 \else%
1366   \AMCif@env{question}{\AMC@error@explain}%
1367 \fi%
1368 }
1369 \newcommand{\explaincontext}[1]{%
1370 \ifAMC@correc%
1371 #1%
1372 \fi%
1373 }

```

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.

```

1374 \def\scoring#1{\ifAMC@calibration\AMCmessage{B=#1}\fi\ignorespaces}
1375 \def\scoringDefaultS#1{\ifAMC@calibration\AMCmessage{BDS=#1}\fi}
1376 \def\scoringDefaultM#1{\ifAMC@calibration\AMCmessage{BDM=#1}\fi}

```

```
1377 \def\QuestionIndicative{\ifAMC@calibration\AMCmessage{INDIC}\fi}
```

4.13 Numerical data

4.13.1 Codes

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

As an example,

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

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

	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
B	3	3	3
C	4	4	4
D	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}`.

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

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

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

```
1378 \newcount\AMC@chiffres
```

```
1379 \newdimen\AMCcodeHspace\AMCcodeHspace=.5em
```

```
1380 \newdimen\AMCcodeVspace\AMCcodeVspace=.5em
```

```
1381 \newcommand\AMCcodeID@squarebrackets[2]{#1[#2]}
```

```
1382 \newcommand\AMCcodeID@dot[2]{#1.#2}
```

```
1383 \newcommand\AMCcodeID@@[1]{%
```

```
1384 \expandafter\def\expandafter\AMCcodeID\expandafter{\csname AMCcodeID@#1\endcsname}%
```

```

1385 \def\AMCcodeID@mode{#1}%
1386 }
1387 \AMCcodeID@@{squarebrackets}
1388 \ExplSyntaxOn
1389
1390 \clist_new:N \amc_code_descr_clist
1391 \seq_new:N \amc_code_digits_seq
1392 \int_new:N \amc_code_digit_n_int
1393 \bool_new:N \amc_code_vertical_bool
1394 \bool_new:N \amc_code_top_bool
1395 \bool_new:N \amc_code_multi_bool
1396 \clist_new:N \amc__multi_clist
1397
1398 \cs_new:Npn \amc_code_init:N #1 {
1399 \def\AMCbeginQuestion##1##2{}
1400 \def\AMCbeforeQuestion{}
1401 \AMCnoScoreZone
1402 \AMCquestionNumberfalse
1403 \setlength{\parindent}{0pt}
1404 \AMCnobloc
1405 \int_set:Nn \amc_code_digit_n_int { \clist_count:N #1 }
1406 }
1407
1408 \cs_new:Nn \amc_code_digit_init: {
1409 \QuestionIndicative
1410 \global\AMCrep@count=\z@
1411 }
1412
1413 \cs_new:Npn \amc_code_digit:n #1 {
1414 \global\advance\AMCrep@count\@ne\relax
1415 \ifAMC@calibration\AMCmessage{ REP = \the\AMCrep@count : M }\fi
1416 \hbox{\AMC@keyBox@{#1}{1}{case : \AMCid@name : \the\AMCid@quest , \the\AMCrep@count}}
1417 \bool_if:NTF \amc_code_vertical_bool {
1418 \vspace{\AMCcodeVspace}
1419 }{
1420 \hspace{\AMCcodeHspace}
1421 }
1422 }
1423
1424 \keys_define:nn { amccode } {
1425 vertical .bool_set:N = \amc_code_vertical_bool,
1426 vertical .initial:n = { true },
1427 vertical .default:n = { true },
1428 v .code:n = { \bool_set_true:N \amc_code_vertical_bool },
1429 h .code:n = { \bool_set_false:N \amc_code_vertical_bool },
1430 top .bool_set:N = \amc_code_top_bool,
1431 top .initial:n = { false },
1432 top .default:n = { true },
1433 multi .bool_set:N = \amc_code_multi_bool,
1434 multi .initial:n = { false },
1435 multi .default:n = { true }
1436 }
1437

```

```

1438 \cs_new_nopar:Nn \amc_multi_report: {
1439   \ifAMC@calibration
1440   \immediate\write\AMC@XYFILE{\string\with{multi=\clist_use:Nn\amc__multi_clist{,}}}
1441   \fi
1442 }
1443 \cs_new_eq:NN \AMC@multi@report \amc_multi_report:
1444 \int_new:N \amc_multi_count_int
1445 \cs_new_nopar:Nn \amc_multi_clear: {
1446   \int_gzero:N \amc_multi_count_int
1447 }
1448 \cs_new_eq:NN \AMC@multiclear \amc_multi_clear:
1449
1450 \cs_new:Npn \amc_code_generate:nNn #1#2#3 {
1451   { \keys_set:nn { amccode } { #3 }
1452     \bool_if:NTF \amc_code_multi_bool {
1453       \clist_gset:Nn \amc__multi_clist { #1 }
1454     } {}
1455     \bool_if:NTF \amc_code_multi_bool { \int_gincr:N \amc_multi_count_int } {}
1456     \amc_code_init:N #2
1457     \clist_map_inline:Nn #2 { % iterates over 'digits'
1458       \begin{question}{
1459         \AMCcodeID{ #1 \bool_if:NTF
1460           \amc_code_multi_bool
1461           { * \int_use:N \amc_multi_count_int } {} }
1462         { \int_use:N \amc_code_digit_n_int }
1463       }
1464       \amc_code_digit_init:
1465       \seq_set_split:Nnn \amc_code_digits_seq {} { ##1 }
1466       \bool_if:NTF \amc_code_vertical_bool {
1467         \hspace{0pt}
1468         \bool_if:NTF \amc_code_top_bool { \vtop } { \vbox }
1469         \bgroup
1470       }{
1471         \hbox\bgroup
1472       }
1473       \seq_map_inline:Nn \amc_code_digits_seq {
1474         % iterates over available characters for 'digit'
1475         \amc_code_digit:n { ####1 }
1476       }
1477       \bool_if:NTF \amc_code_vertical_bool {
1478         \vspace{-\AMCcodeVspace}\egroup
1479         \hspace{\AMCcodeHspace}
1480       }{
1481         \egroup\vspace{\AMCcodeVspace}
1482         \par
1483       }
1484       \end{question}
1485       \int_decr:N \amc_code_digit_n_int
1486     }
1487   }
1488 }
1489
1490 \cs_new:Npn \amc_code_generate:nnn #1#2#3 {

```

```

1491 \clist_set:Nn \amc_code_descr_clist { #2 }
1492 \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1493 }
1494 \cs_generate_variant:Nn \amc_code_generate:nnn { xxx }
1495 \newcommand{\AMCcodeGrid}[3][ ]{
1496   \amc_code_generate:xxx { #2 } { #3 } { #1 }
1497 }
1498
1499 \cs_new:Npn \amc_code_generate_integer:nnn #1#2#3 {
1500   \clist_clear:N \amc_code_descr_clist
1501   \prg_replicate:nn { #2 } { \clist_put_right:Nn \amc_code_descr_clist { 0123456789 } }
1502   \amc_code_generate:nNn { #1 } \amc_code_descr_clist { #3 }
1503 }
1504 \cs_generate_variant:Nn \amc_code_generate_integer:nnn { xxx }
1505 \newcommand{\AMCcodeGridInt}[3][ ]{
1506   \amc_code_generate_integer:xxx { #2 } { #3 } { #1 }
1507 }
1508
1509 \cs_new:Npn \amc_code_generate_integer_v:nn #1#2 {
1510   \amc_code_generate_integer:nnn { #1 } { #2 } { v }
1511 }
1512 \cs_new:Npn \amc_code_generate_integer_h:nn #1#2 {
1513   \amc_code_generate_integer:nnn { #1 } { #2 } { h }
1514 }
1515 \cs_generate_variant:Nn \amc_code_generate_integer_v:nn { xx }
1516 \cs_generate_variant:Nn \amc_code_generate_integer_h:nn { xx }
1517 \cs_new_eq:NN \AMCcode \amc_code_generate_integer_v:xx
1518 \cs_new_eq:NN \AMCcodeH \amc_code_generate_integer_h:xx
1519
1520 \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 ?
☐0 ☐1 ☐2 ☒3 ☐4 ☐5 ☐6 ☐7 ☐8 ☐9 ☐0 ☐1 ☐2 ☐3 ☐4 ☒5 ☐6 ☐7 ☐8 ☐9	

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

This command uses the `\AMCformatChoices{<showcommand>}{<hidecommand>}{<arg1>}{<arg2>}` command, that calls either `<hidecommand>{<arg1>}{<arg2>}` if the `separateanswersheet` option is used and if we are currently in the question part (not in the answer sheet), or `<showcommand>{<arg1>}{<arg2>}` when all the boxes are to be produced.

```

1521 \newcommand\AMCformatChoices[4]{%
1522   \global\AMCrep@count=\z@%
1523   \AMC@if@separate@question{%
1524     \AMC@mem@add{\global\AMCrep@count=\z@%
1525               #1{#3}{#4}}%
1526   }%
1527   \ifAMC@ensemble%
1528     #2{#3}{#4}%
1529     \AMCmessage{QPART}%
1530   \else%
1531     #1{#3}{#4}%
1532   \fi%
1533 }
```

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:Nnn{<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.

```

1534 \ExplSyntaxOn
1535
1536 \cs_generate_variant:Nn \tl_replace_once:Nnn { Nxn }
1537
1538 \tl_new:N \amc_ee_tl
1539 \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.

```

1540 \group_begin:
1541 \char_set_catcode_other:N E
1542 \tex_lowercase:D
1543 {
1544   \cs_new:Npn \amc_read_scientific:NNn #1 #2 #3 {
1545     \tl_set:Nn \amc_ee_tl { #3 }
1546     \tl_replace_once:Nxn \amc_ee_tl { E } { e }
1547     \seq_set_split:NnV \amc_ee_seq e \amc_ee_tl
1548     \fp_set:Nn #1 { \seq_item:Nn \amc_ee_seq 1 }
1549     \int_set:Nn #2 { \seq_item:Nn \amc_ee_seq 2 }
1550   }
1551 }
1552 \group_end:
1553
1554 \cs_generate_variant:Nn \amc_read_scientific:NNn { NNf, NNx }
1555
1556 \fp_new:N \amc_fulls_fp
1557 \cs_new:Npn \amc_fp_decompose:NNn #1 #2 #3 {
1558   \fp_set:Nn \amc_fulls_fp { #3 }
```

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

```

1559 \amc_read_scientific:NNx #1 #2
1560 { \fp_to_scientific:N \amc_fulls_fp e 0 }
1561 }
1562 \cs_generate_variant:Nn \amc_fp_decompose:NNn { NNx }
1563
1564 \fp_new:N \amc_num_mantissa_fp
1565 \int_new:N \amc_num_exponent_int
1566 \cs_new:Npn \amc_fp_n_significant_digits:Nnn #1 #2 #3 {
1567   \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_exponent_int
1568   { #2 }
1569   \fp_set:Nn #1
1570   { round(\amc_num_mantissa_fp * 10^((#3)-1)) }
1571   \fp_compare:nTF { abs(#1) >= 10^(#3) }
1572   {
1573     \fp_set:Nn #1 { #1 / 10 }
1574   } { }
1575 }
1576
1577 \fp_new:N \amc_num_nsig_fp
1578 \cs_new:Npn \amc_fp_show_n_significant_digits:nn #1 #2 {
1579   \amc_fp_n_significant_digits:Nnn \amc_num_nsig_fp { #1 } { #2 }
1580 }
1581 \cs_new_eq:NN \AMCsignificantDigits \amc_fp_show_n_significant_digits:nn
1582
1583 \cs_new:Npn \amc_fp_show_significant_digits: {
1584   \fp_use:N \amc_num_nsig_fp
1585 }
1586 \cs_new_eq:NN \AMCshowSignificantDigits \amc_fp_show_significant_digits:
1587
1588 \cs_new:Npn \amc_fp_n_digits:Nnn #1 #2 #3 {
1589   \fp_set:Nn #1
1590   { round((#2) * \amc_num_base_int^(#3)) }
1591 }
1592
1593 \int_new:N \amc_todigits_int
1594 \cs_new:Npn \amc_fp_to_digits:Nnnn #1 #2 #3 #4 {
1595   \clist_clear:N #1
1596   \int_set:Nn \amc_todigits_int { \fp_eval:n { abs(round(#2)) } }
1597   \prg_replicate:nn { #3 } {
1598     \clist_put_left:Nx #1 { \int_mod:nn \amc_todigits_int { #4 } }
1599     \int_set:Nn \amc_todigits_int
1600     { \int_div_truncate:nn \amc_todigits_int { #4 } }
1601   }
1602   \int_compare:nNnTF \amc_todigits_int = 0 { } {
1603     \message{^^J!~Error~(question~\AMCid@name):~number~too~large,~some~digits~will~be~discarded^^J}
1604   }
1605 }
1606
1607 \cs_new:Npn \amc_invalid_digits:Nn #1 #2 {
1608   \clist_clear:N #1
1609   \prg_replicate:nn { #2 } { \clist_put_left:Nx #1 { -1 } }

```

```

1610 }
1611
1612 \cs_new:Npn \amc_get_fp_sign:Nn #1 #2 {
1613   \fp_compare:nNnTF #2 < 0 {
1614     \int_set:Nn #1 { -1 }
1615   }{
1616     \fp_compare:nNnTF #2 > 0 {
1617       \int_set:Nn #1 { 1 }
1618     }{
1619       \int_set:Nn #1 { 0 }
1620     }
1621   }
1622 }
1623
1624 \cs_new:Npn \amc_get_int_sign:Nn #1 #2 {
1625   \int_compare:nNnTF #2 < 0 {
1626     \int_set:Nn #1 { -1 }
1627   }{
1628     \int_compare:nNnTF #2 > 0 {
1629       \int_set:Nn #1 { 1 }
1630     }{
1631       \int_set:Nn #1 { 0 }
1632     }
1633   }
1634 }
1635
1636 \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:

```

1637 \def\AMCnntextGoto{}
1638 \def\AMCnntextVHead#1{\emph{b#1}}
1639 \newdimen\AMCnumeric@Hspace\AMCnumeric@Hspace=.5em
1640 \newdimen\AMCnumeric@Vspace\AMCnumeric@Vspace=1ex
1641 \ExplSyntaxOn
1642
1643 \keys_define:nn { amcnumeric } {
1644   Tsign .code:n = {\def\AMCnntextSign{#1}},
1645   Tsign .initial:n = {},
1646   Tpoint .code:n = {\def\AMCdecimalPoint{#1}},
1647   Tpoint .initial:n = { \raisebox{1ex}{\bf .} },
1648   Texponent .code:n = {\def\AMCexponent{#1}},
1649   Texponent .initial:n = { $\times 10^{\textasciicircum{}}$ },
1650   vspace .code:n = {\AMCnumeric@Vspace=#1},
1651   hspace .code:n = {\AMCnumeric@Hspace=#1},
1652   arraybordercol .code:n = {\def\AMCnarray@Border{#1}},
1653   arraybordercol .initial:n = { black ! 20 },
1654   arrayborderwidth .code:n = {\def\AMCnarray@BorderWidth{#1}},
1655   arrayborderwidth .initial:n = { .5mm },
1656   bordercol .code:n = {\def\AMCncol@Border{#1}},
1657   bordercol .initial:n = { lightgray },
1658   borderwidth .code:n = {\def\AMCncol@BorderWidth{#1}},

```

```

1659 borderwidth .initial:n = { 1mm },
1660 backgroundcol .code:n = {\def\AMCncol@Background{#1}},
1661 backgroundcol .initial:n = { white },
1662 digits .int_set:N = \amc_num_ndigits_int,
1663 digits .initial:n = { 3 },
1664 digitcols .int_set:N = \amc_num_digitcols_int,
1665 digitcols .initial:n = { 1 },
1666 decimals .int_set:N = \amc_num_decd_int,
1667 decimals .initial:n = { 0 },
1668 exponent .int_set:N = \amc_num_expo_int,
1669 exponent .initial:n = { 0 },
1670 base .int_set:N = \amc_num_base_int,
1671 base .initial:n = { 10 },
1672 sign .bool_set:N = \amc_num_sign_bool,
1673 sign .initial:n = { true },
1674 sign .default:n = { true },
1675 plussign .bool_set:N = \amc_num_plussign_bool,
1676 plussign .initial:n = { true },
1677 plussign .default:n = { true },
1678 exposign .bool_set:N = \amc_num_exposign_bool,
1679 exposign .initial:n = { true },
1680 exposign .default:n = { true },
1681 strict .bool_set:N = \amc_num_strict_bool,
1682 strict .initial:n = { false },
1683 strict .default:n = { true },
1684 scoring .bool_set:N = \amc_num_scoring_bool,
1685 scoring .initial:n = { true },
1686 scoring .default:n = { true },
1687 ignoreblank .bool_set:N = \amc_num_ignoreblank_bool,
1688 ignoreblank .initial:n = { false },
1689 ignoreblank .default:n = { true },
1690 vertical .bool_set:N = \amc_num_vertical_bool,
1691 vertical .initial:n = { false },
1692 vertical .default:n = { true },
1693 expovertical .bool_set:N = \amc_num_expovertical_bool,
1694 expovertical .initial:n = { false },
1695 expovertical .default:n = { true },
1696 reverse .bool_set:N = \amc_num_reverse_bool,
1697 reverse .initial:n = { false },
1698 reverse .default:n = { true },
1699 vhead .bool_set:N = \amc_num_vhead_bool,
1700 vhead .initial:n = { false },
1701 vhead .default:n = { true },
1702 Tvhead .code:n = {\clist_set:Nx \amc_tvhead_clist {\clist_reverse:n {#1}}},
1703 Tvhead .initial:n = {}, % \c_empty_clist does not work with \clist_reverse:n
1704 vheadunitindex .int_set:N = \amc_vheadunitindex_int,
1705 vheadunitindex .initial:n = 0,
1706 nozero .bool_set:N = \amc_num_nozero_bool,
1707 nozero .initial:n = { false },
1708 nozero .default:n = { true },
1709 significant .bool_set:N = \amc_num_significant_bool,
1710 significant .initial:n = { false },
1711 significant .default:n = { true },

```

```

1712 scoreexact .code:n = {\def\AMC@numeric@scoreexact{#1}},
1713 scoreexact .initial:n = { 2 },
1714 scoreapprox .code:n = {\def\AMC@numeric@scoreapprox{#1}},
1715 scoreapprox .initial:n = { 1 },
1716 scorewrong .code:n = {\def\AMC@numeric@scorewrong{#1}},
1717 scorewrong .initial:n = { 0 },
1718 exact .int_set:N = \amc_num_exact_int,
1719 exact .initial:n = { 0 },
1720 approx .int_set:N = \amc_num_approx_int,
1721 approx .initial:n = { 0 },
1722 keepas .code:n = {\def\AMC@numeric@keepas{#1}},
1723 keepas .initial:n = {},
1724 alsocorrect .code:n = {\def\AMC@numeric@alsocorrect{#1}},
1725 alsocorrect .initial:n = {}
1726 }
1727
1728 \cs_new:Npn \amc_num_setopts #1 {
1729   \keys_set:nn { amcnumeric } { #1 }
1730 }
1731
1732 \cs_new_nopar:Nn \amc_num_check_score_opts: {
1733   \bool_if:NTF \amc_num_ignoreblank_bool {
1734     \int_compare:nNnTF \amc_num_base_int = { 10 } { } {
1735       \message{^^J!~Error:~ignoreblank~can~only~be~used~with~number~base~10^^J}
1736     }
1737   } {}
1738 }
1739
1740 \cs_new_eq:NN \AMCnumericOpts \amc_num_setopts
1741

```

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.

```

1742 \cs_new:Npn \amc_num_char:nn #1 #2 {
1743   \global\advance\AMCrep@count\@ne\relax
1744   \AMCmessage{REP= \the\AMCrep@count :
1745     \ifx#2\AMC@checkbox B\else M\fi }
1746   \ifAMC@correc
1747     \protect\AMC@keyBox@{#1}{#2}{1}{case : \AMCid@name :
1748       \the\AMCid@quest , \the\AMCrep@count}
1749   \else
1750     \protect\AMC@keyBox@{#1}{-}{1}{case : \AMCid@name :
1751       \the\AMCid@quest , \the\AMCrep@count}
1752   \fi
1753 }

```

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

```

1754 \int_new:N \amc_num_digit_value_int
1755 \tl_new:N \amc_num_digit_value_tl
1756 \cs_new:Npn \amc_num_digit_box:nn #1 #2 {
1757   \int_set:Nn \amc_num_digit_value_int { #1 }

```

```

1758 \tl_set:Nn \amc_num_digit_value_tl {
1759   \int_compare:nNnTF { \amc_num_digit_value_int } < { 10 }
1760   { \int_to_arabic:n { \amc_num_digit_value_int } }
1761   { \int_to_Alph:n { \amc_num_digit_value_int - 9 } }
1762 }
1763 \int_compare:nNnTF { #1 } = { #2 } {
1764   \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1765   {\AMC@checkbox}
1766 } {
1767   \amc_num_char:nn{ \tl_use:N \amc_num_digit_value_tl }
1768   {}
1769 }
1770 }

```

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

```

1771 \cs_new:Npn \amc_num_sign_boxes:Nn #1 #2 {
1772   \int_case:nn { #1 } {
1773     { -1 } {
1774       \bool_if:NTF \amc_num_plussign_bool {
1775         \hbox{\amc_num_char:nn{${+}$}{}}
1776         \vspace{\AMCnumeric@Vspace}
1777         \AMCmessage{B=set. sign #2 =1}
1778       } { }
1779       \hbox{\amc_num_char:nn{${-}$}{\AMC@checkbox}}
1780       \AMCmessage{B=set. sign #2 =-1}
1781     }
1782     { 1 } {
1783       \bool_if:NTF \amc_num_plussign_bool {
1784         \hbox{\amc_num_char:nn{${+}$}{\AMC@checkbox}}
1785         \vspace{\AMCnumeric@Vspace}
1786         \AMCmessage{B=set. sign #2 =1}
1787       } { }
1788       \hbox{\amc_num_char:nn{${-}$}{}}
1789       \AMCmessage{B=set. sign #2 =-1}
1790     }
1791     { 0 } {
1792       \bool_if:NTF \amc_num_plussign_bool {
1793         \hbox{\amc_num_char:nn{${+}$}{}}
1794         \vspace{\AMCnumeric@Vspace}
1795         \AMCmessage{B=set. sign #2 =1}
1796       } { }
1797       \hbox{\amc_num_char:nn{${-}$}{}}
1798       \AMCmessage{B=set. sign #2 =-1}
1799     }
1800   }
1801 }

```

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

```

1802 \cs_new:Npn \amc_num_digit_boxes_h:nnn #1 #2 #3 {
1803   \int_step_inline:nnnn

```

```

1804 { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1805 { 1 } { #3 - 1 } {
1806   \amc_num_digit_box:nn { ##1 } { #2 }
1807   \AMCmessage{B= set. #1 = ##1}
1808   \hspace{\AMCnumeric@Hspace}
1809 }
1810 \hspace{-\AMCnumeric@Hspace}
1811 }
1812

```

The command `\amc_num_digit_boxes_v:nnn{<varname>}{<correct>}{<maxdigit>}` does the same as `\amc_num_digit_boxes_h:nnn`, with a vertical disposition.

```

1813 \cs_new:Npn \amc_num_digit_boxes_v:nnn #1 #2 #3 {
1814   \int_step_inline:nnnn
1815   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1816   { 1 } { #3 - 1 } {
1817     \vbox{\hbox{
1818       \amc_num_digit_box:nn { ##1 } { #2 }
1819     }}
1820     \AMCmessage{B= set. #1 = ##1}
1821     \int_compare:nNnTF { ##1 } < { #3 - 1 } {
1822       \vspace{\AMCnumeric@Vspace}
1823     } {}
1824   }
1825 }
1826

```

The command `\amc_num_digit_boxes_a:nnn{<varname>}{<correct>}{<maxdigit>}` does the same as `\amc_num_digit_boxes_h:nnn`, with an array disposition with `\amc_num_digitcols_int` columns.

```

1827 \int_new:N \amc_num_array_int
1828 \int_new:N \amc_num_array_col_int
1829 \int_new:N \amc_num_array_row_int
1830 \int_new:N \amc_num_array_nrows_int
1831 \int_new:N \amc_num_array_i_int
1832 \tl_new:N \amc_num_array_tl
1833
1834 \cs_new:Npn \amc_tabular_defs:nn #1 #2 {
1835   \begin{tabular}[#1]{#2}
1836 }
1837 \cs_generate_variant:Nn \amc_tabular_defs:nn { xx }
1838
1839 \cs_new:Npn \amc_num_digit_boxes_a:nnn #1 #2 #3 #4 {
1840   \tl_set:Nn \amc_num_array_tl 1
1841   \int_step_inline:nnnn 2 1 \amc_num_digitcols_int {
1842     \tl_put_right:Nn \amc_num_array_tl { |1 }
1843   }
1844   \fboxrule=\AMCnarray@BorderWidth
1845   \fboxsep=1pt
1846   \fcolorbox{\AMCnarray@Border}{\AMCncol@Background}{\hspace*{-.2em}
1847     \arrayrulecolor{\AMCnarray@Border}\setlength{\tabcolsep}{4pt}
1848     \hbox{\amc_tabular_defs:xx { b } { \tl_use:N \amc_num_array_tl }
1849     \int_gzero:N \amc_num_array_int
1850

```

```

1851 \int_gzero:N \amc_num_array_row_int
1852 \int_gset:Nn \amc_num_array_nrows_int
1853 { \int_div_truncate:nn { #4 - #3 + \amc_num_digitcols_int - 1 }
1854   \amc_num_digitcols_int }
1855 \int_do_while:nNnn \amc_num_array_row_int < \amc_num_array_nrows_int {
1856   \int_gzero:N \amc_num_array_col_int
1857   \int_do_while:nNnn \amc_num_array_col_int < \amc_num_digitcols_int {
1858
1859     \int_set:Nn \amc_num_array_i_int
1860     { #3 + \amc_num_array_row_int +
1861       \amc_num_array_col_int * \amc_num_array_nrows_int }
1862
1863     \int_compare:nNnTF \amc_num_array_i_int < #4 {
1864       \vbox{\hbox{
1865         \amc_num_digit_box:nn { \int_use:N \amc_num_array_i_int }{ #2 }
1866       }}
1867       \AMCmessage{B= set. #1 = \int_use:N \amc_num_array_i_int}
1868     } { }
1869
1870     \int_gincr:N \amc_num_array_col_int
1871
1872     \int_compare:nNnTF \amc_num_array_col_int < \amc_num_digitcols_int { & } { }
1873   }
1874   \int_gincr:N \amc_num_array_row_int
1875
1876   \int_compare:nNnTF \amc_num_array_row_int < \amc_num_array_nrows_int { \\ } { }
1877 }
1878
1879 \end{tabular}}\hspace*{-.2em}}
1880 }
1881
1882 \int_new:N \amc_num_first_digit_int
1883 \cs_new:Npn \amc_num_digit_boxes_vr:nnn #1 #2 #3 {
1884   \int_set:Nn \amc_num_first_digit_int
1885   { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1886   \int_step_inline:nnnn { #3 - 1 } { -1 }
1887   \amc_num_first_digit_int {
1888     \vbox{\hbox{
1889       \amc_num_digit_box:nn { ##1 }{ #2 }
1890     }}
1891     \AMCmessage{B= set. #1 = ##1}
1892     \int_compare:nNnTF { ##1 } > \amc_num_first_digit_int {
1893       \vspace{\AMCnumeric@Vspace}
1894     } {}
1895   }
1896 }

The command \amc_num_integer_boxes_v:Nnn{<correct digits>}{<prefix>}{<decimals>} draws
boxes for integer entry, without the sign.
1897 \cs_new:Npn \amc_num_integer_boxes_v:Nnn #1 #2 #3 {
begin a loop over all digits,
1898   \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1899   \clist_map_inline:Nn #1 {

```

place the decimal point if necessary,

```
1900 \int_compare:nNnTF \amc_num_digit_int = { #3 } {
1901   \hbox{ \AMCdecimalPoint }\hspace{\AMCnumeric@Hspace}
1902 } { }
```

draw the box for this digit,

```
1903 \hbox{\vbox{
1904   \bool_if:NTF \amc_num_vhead_bool {
1905     \vbox{\hbox to \AMC@boxedwidth{\hfill\AMCnntextVHead{ \int_eval:n
1906       { \amc_num_digit_int - 1 } }\hfill}}
1907     \vspace{\AMCnumeric@Vspace}
1908   } { }
1909   \int_compare:nNnTF \amc_num_digitcols_int = 1 {
1910     \bool_if:NTF \amc_num_reverse_bool {
1911       \amc_num_digit_boxes_vr:nnn { #2
1912         \int_to_Alph:n \amc_num_digit_int }
1913       { ##1 } { \amc_num_base_int }
1914     } {
1915       \amc_num_digit_boxes_v:nnn { #2
1916         \int_to_Alph:n \amc_num_digit_int }
1917       { ##1 } { \amc_num_base_int }
1918     }
1919   } {
1920     \hbox{
1921       \amc_num_digit_boxes_a:nnn { #2
1922         \int_to_Alph:n \amc_num_digit_int }
1923       { ##1 } { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } }
1924       \amc_num_base_int
1925     }
1926   }
1927 }}
```

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

```
1928 \int_compare:nNnTF \amc_num_digit_int > 1 {
1929   \hspace{\AMCnumeric@Hspace}
1930 } { }
1931 \int_decr:N \amc_num_digit_int
1932 }
1933 }
1934
```

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

```
1935
1936 \cs_new:Npn \amc_num_integer_boxes_h:Nnn #1 #2 #3 {
1937   \vbox{
1938     \int_set_eq:NN \amc_num_digit_int { \clist_count:N #1 }
1939     \clist_map_inline:Nn #1 {
1940       \int_compare:nNnTF
1941       \amc_num_digit_int = { #3 } {
1942         \hbox{ \AMCdecimalPoint }
1943       } { }
1944       \int_compare:nNnTF \amc_num_digitcols_int = 1 {
1945         \hbox{
```

```

1946         \amc_num_digit_boxes_h:nnn { #2
1947             \int_to_Alph:n \amc_num_digit_int }
1948         { ##1 } \amc_num_base_int
1949     }
1950 } {
1951     \hbox{
1952         \amc_num_digit_boxes_a:nnn { #2
1953             \int_to_Alph:n \amc_num_digit_int }
1954         { ##1 } { \bool_if:NTF \amc_num_nozero_bool { 1 } { 0 } } \amc_num_base_int
1955     }
1956 }
1957 \int_compare:nNnTF \amc_num_digit_int > 1 {
1958     \vspace{\AMCnumeric@Vspace}
1959 } { }
1960 \int_decr:N \amc_num_digit_int
1961 }}
1962 }
1963

```

```

1964 \cs_new:Npn \amc_num_sign_boxes:NnN #1 #2 #3 {
1965     \vbox{
1966         \ifx\AMCnntextSign\empty\empty\else
1967             \hbox{\AMCnntextSign}\vspace{\AMCnumeric@Vspace}\fi
1968         \bool_if:NTF \amc_num_strict_bool {
1969             \int_compare:nNnTF { #3 } = 0 {
1970                 \amc_num_sign_boxes:Nn { 1 } { #2 }
1971             }{
1972                 \amc_num_sign_boxes:Nn { #3 } { #2 }
1973             }
1974         }{
1975             \amc_num_sign_boxes:Nn { #3 } { #2 }
1976         }
1977     }
1978 }
1979

```

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.

```

1980 \cs_new:Npn \amc_num_integer_boxes:NnnNN #1 #2 #3 #4 #5 {
1981     \hbox{
1982         \bool_if:NTF { #4 } {
1983             \int_compare:nNnTF \amc_num_digitcols_int = 1 {
1984                 \amc_num_sign_boxes:NnN { #1 } { #2 } { #5 }
1985                 \hspace{\AMCnumeric@Hspace}
1986                 \vrule
1987                 \hspace{\AMCnumeric@Hspace}
1988             } {
1989                 \fboxrule=\AMCnarray@BorderWidth
1990                 \fboxsep=1pt
1991                 \fcolorbox{\AMCnarray@Border}{\AMCncol@Background}{
1992                     \amc_num_sign_boxes:NnN { #1 } { #2 } { #5 }
1993                 }
1994                 \hspace{\AMCnumeric@Hspace}
1995             }
1996         }
1997     }
1998 }

```

```

1996   } { }
1997   \hbox{
1998     \bool_if:NTF \amc_num_vertical_bool
1999     \amc_num_integer_boxes_v:Nnn \amc_num_integer_boxes_h:Nnn
2000     #1 { #2 } { #3 }
2001   }
2002 }
2003 }
2004

```

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

```

2005
2006 \cs_new:Npn \amc_num_build_integer_scoring:Nnnnn #1 #2 #3 #4 #5 {
2007   \amc_num_check_score_opts:
2008   \tl_clear:N #1
2009   \int_set_eq:NN \amc_num_digit_int { #4 }
2010   \int_while_do:nNnn \amc_num_digit_int > 0 {
2011     \bool_if:NTF \amc_num_strict_bool {
2012       \AMCmessage{B=requires. #3
2013         \int_to_Alph:n \amc_num_digit_int = 1}
2014     } {
2015       \AMCmessage{B=default. #3
2016         \int_to_Alph:n \amc_num_digit_int =
2017         \bool_if:NTF \amc_num_ignoreblank_bool { } { 0 }
2018       }
2019     }
2020     \int_compare:nNnTF \amc_num_digit_int = #4 { } {
2021       \bool_if:NTF \amc_num_ignoreblank_bool {
2022         \tl_put_right:Nx #1 { ~.~ }
2023         \int_compare:nNnTF \amc_num_digit_int = #5 {
2024           \tl_put_right:Nx #1 { "." ~.~ }
2025         } { }
2026       } {
2027         \tl_put_left:Nn #1 { ( }
2028         \tl_put_right:Nx #1 { ) } *
2029         \int_use:N \amc_num_base_int + }
2030     }
2031   }
2032   \tl_put_right:Nx #1
2033   { #3 \int_to_Alph:n \amc_num_digit_int }
2034   \int_decr:N \amc_num_digit_int
2035 }
2036 \bool_if:NTF \amc_num_ignoreblank_bool {
2037   \tl_put_left:Nn #1 { ( 0 + ( }
2038   \tl_put_right:Nn #1 { ) ) }
2039   \int_compare:nNnTF \amc_num_decd_int > 0 {
2040     \tl_put_right:Nx #1 { * ( \int_use:N \amc_num_base_int ** \int_eval:n { #5 } ) }
2041   } { }
2042 } {
2043   \tl_put_left:Nn #1 { ( }
2044   \tl_put_right:Nn #1 { ) }
2045 }

```

```

2046 \bool_if:NTF { #2 } {
2047   \bool_if:NTF \amc_num_strict_bool {
2048     \AMCmessage{B=requires. sign #3 =1}
2049   } {
2050     \AMCmessage{B=default. sign #3 =1}
2051   }
2052   \tl_put_right:Nx #1 { * ( sign #3 ) }
2053 } { }
2054 }
2055

```

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

```

2056
2057 \fp_new:N \amc_num_result_fp
2058 \fp_new:N \amc_num_correct_fp
2059 \clist_new:N \amc_num_digits_clist
2060 \clist_new:N \amc_num_expo_digits_clist
2061 \int_new:N \amc_num_digit_int
2062 \int_new:N \amc_num_sign_int
2063 \int_new:N \amc_num_expo_sign_int
2064 \tl_new:N \amc_num_compute_tl
2065 \tl_new:N \amc_num_expo_tl
2066 \int_new:N \amc_num_correct_expo_int
2067
2068 \cs_new:Npn \amc_numeric_show:nn #1 #2 {

```

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

```

2069 \ifAMC@ensemble\ifAMCformulaire@dedans
2070   \AMCmessage{Q=\the\AMCid@quest}
2071 \fi\fi

```

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

```

2072 {\keys_set:nn { amcnumeric } { #2 }

```

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

```

2073 \int_compare:nTF {\amc_vheadunitindex_int = 0}
2074 {\int_set:Nn \amc_vheadunitindex_int {\amc_num_decd_int + 1}}
2075 {}
2076
2077 \clist_if_empty:NTF \amc_tvhead_clist
2078 {}
2079 {
2080   \def\AMCncontextVHead##1{
2081     \int_set:Nn \l_tmpa_int
2082       {\int_max:nn
2083         {((##1) - \amc_num_decd_int + \amc_vheadunitindex_int)
2084         {0}}}
2085     \emph{\clist_item:Nn \amc_tvhead_clist {\l_tmpa_int}}}}
2086
2087   \bool_if:nTF { \bool_if_p:N\amc_num_significant_bool
2088     && \int_compare_p:n { \amc_num_base_int != 10 } } {
2089     \message{^^J!~AMCnumeric~Error:~significant=true~can't~be~used~with~base!=10.^^J}
2090   } {}
2091   \bool_if:nTF { \int_compare_p:n { \amc_num_expo_int != 0 }
2092     && \int_compare_p:n { \amc_num_base_int != 10 } } {

```

```

2093     \message{^^J!~AMCnumeric~Error:~scientific~notation~can't~be~used~with~base!=10.^^J}
2094 } {}

```

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

```

2095 \ifx\@empty#1\@empty
2096   \fp_set:Nn \amc_num_correct_fp { 0 }
2097   \fp_set:Nn \amc_num_mantissa_fp { 0 }
2098   \int_set:Nn \amc_num_correct_expo_int { 0 }
2099 \else
2100 \bool_if:NTF \amc_num_significant_bool {
2101   \amc_fp_n_significant_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_ndigits_int
2102 } {
2103   \int_compare:nNnTF \amc_num_expo_int > 0 {
2104     \amc_fp_decompose:NNn \amc_num_mantissa_fp \amc_num_correct_expo_int { #1 }
2105     \int_compare:nNnTF { \amc_num_ndigits_int - \amc_num_decd_int } > 1 {
2106       \fp_set:Nn \amc_num_mantissa_fp {
2107         \amc_num_mantissa_fp * \amc_num_base_int^( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
2108       }
2109       \int_set:Nn \amc_num_correct_expo_int {
2110         \amc_num_correct_expo_int - ( \amc_num_ndigits_int - \amc_num_decd_int - 1 )
2111       }
2112     } {}
2113   } \amc_fp_n_digits:Nnn \amc_num_correct_fp \amc_num_mantissa_fp \amc_num_decd_int

```

In scientific mode, an adjustment has to be done if eg. 9.999e10 has been rounded to 10.0e10: we must replace it by 1.0e11.

```

2114   \fp_compare:nTF { \amc_num_correct_fp >= \amc_num_base_int^( \amc_num_ndigits_int ) }
2115   {
2116     \fp_set:Nn \amc_num_correct_fp { round ( \amc_num_correct_fp / \amc_num_base_int ) }
2117     \int_set:Nn \amc_num_correct_expo_int { \amc_num_correct_expo_int + 1 }
2118   } {}
2119 } {
2120   \amc_fp_n_digits:Nnn \amc_num_correct_fp { #1 } \amc_num_decd_int
2121 }
2122 }
2123 \fi

```

Now extracts the required digits:

```

2124 \ifx\@empty#1\@empty
2125   \amc_invalid_digits:Nn \amc_num_digits_clist \amc_num_ndigits_int
2126   \amc_invalid_digits:Nn \amc_num_expo_digits_clist \amc_num_expo_int
2127   \int_set:Nn \amc_num_sign_int { 0 }
2128   \int_set:Nn \amc_num_expo_sign_int { 0 }
2129 \else
2130   \amc_fp_to_digits:Nnnn \amc_num_digits_clist \amc_num_correct_fp
2131   \amc_num_ndigits_int \amc_num_base_int
2132   \amc_get_fp_sign:Nn \amc_num_sign_int \amc_num_correct_fp
2133   \int_compare:nNnTF \amc_num_expo_int > 0 {
2134     \amc_fp_to_digits:Nnnn \amc_num_expo_digits_clist \amc_num_correct_expo_int
2135     \amc_num_expo_int \amc_num_base_int
2136     \amc_get_int_sign:Nn \amc_num_expo_sign_int \amc_num_correct_expo_int
2137   } {}
2138 \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.

```

2139 \fp_set:Nn \amc_num_result_fp { #1 }
2140 \AMCmessage{B=numval=\fp_to_scientific:N \amc_num_result_fp ,
2141   numex=\int_use:N \amc_num_exact_int,
2142   numapp=\int_use:N \amc_num_approx_int,
2143   numsex=\AMC@numeric@scoreexact,
2144   numsapp=\AMC@numeric@scoreapprox
2145 }
2146 \bool_if:NTF \amc_num_scoring_bool {
2147   \AMCmessage{B=haut=mz=d=undef,p=undef,
2148     formula=(Vdifference <= \int_use:N \amc_num_exact_int ?
2149     \AMC@numeric@scoreexact :
2150     \int_compare:nNnTF \amc_num_approx_int = 0 {
2151       \AMC@numeric@scorewrong
2152     } {
2153       (Vdifference <= \int_use:N\amc_num_approx_int ?
2154       \AMC@numeric@scoreapprox : \AMC@numeric@scorewrong)
2155     }
2156   }}
2157 } {}
2158 \amc_num_build_integer_scoring:Nnnnn
2159 \amc_num_compute_tl \amc_num_sign_bool { digit } \amc_num_ndigits_int
2160 \amc_num_decd_int
2161 \int_compare:nNnTF \amc_num_expo_int > 0 {
2162   \amc_num_build_integer_scoring:Nnnnn
2163   \amc_num_expo_tl \amc_num_exposign_bool { expo } \amc_num_expo_int { 0 }
2164   \AMCmessage{B= set. intE = \amc_num_expo_tl}
2165 } {}
2166 \AMCmessage{B= set.intV = \fp_to_int:N\amc_num_correct_fp ,
2167   set.intXX = \amc_num_compute_tl }
2168 \int_compare:nNnTF \amc_num_expo_int > 0 {
2169   \AMCmessage{B= set.intX = intXX * \int_use:N\amc_num_base_int ** ( intE - (\int_use:N\amc_num_correc
2170 }{
2171   \AMCmessage{B= set.intX = intXX}
2172 }
2173 \int_compare:nNnTF \amc_num_expo_int > 0 {
2174   \AMCmessage{B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (intE - \int_use:N\amc_num_decd
2175 }{
2176   \AMCmessage{B= set.valueX = intXX * \int_use:N\amc_num_base_int ** (- \int_use:N\amc_num_decd_int))
2177 }
2178 \ifx\@empty\AMC@numeric@keepas\@empty\else
2179 \AMCmessage{B= setglobal.\AMC@numeric@keepas = valueX}
2180 \fi
2181 \ifx\@empty#1\@empty
2182 \bool_if:NTF \amc_num_significant_bool {
2183   \AMCmessage{B=set.Vdifference=0}
2184 }{
2185   \ifx\@empty\AMC@numeric@alsocorrect\@empty
2186     \AMCmessage{B=set.Vdifference=0}
2187   \else
2188     \AMCmessage{B="set.Vdifference =
2189     amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\amc_r

```

```

2190     "}"
2191   \fi
2192 }
2193 \else
2194 \bool_if:NTF \amc_num_significant_bool {
2195   \AMCmessage{B=set.Vdifference="min( abs((intV)-(intX)) ,
2196     abs(\int_use:N\amc_num_base_int * (intV) - (intX)) ,
2197     abs((intV) - \int_use:N\amc_num_base_int * (intX)) )"}
2198 } {
2199   \ifx\@empty\AMC@numeric@alsocorrect\@empty
2200   \AMCmessage{B=set.Vdifference=abs((intV)-(intX))}
2201   \else
2202   \AMCmessage{B="set.Vdifference =
2203     min( amcvdifference( \AMC@numeric@alsocorrect, valueX, \int_use:N\amc_num_decd_int, \int_use:N\am
2204     abs((intV)-(intX)) )"}
2205   \fi
2206 }
2207 \fi

```

Begin now with the frame around all the boxes:

```

2208 \ifAMC@extractOnly\else
2209 \vspace{1.5ex}\par{
2210   \fboxrule=\AMCncol@BorderWidth
2211   \fcolorbox{\AMCncol@Border}{\AMCncol@Background}{
2212     \bool_if:NTF \amc_num_expovetical_bool {
2213       \hbox{\vbox{
2214         \vbox{\amc_num_integer_boxes:NnnNN
2215           \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
2216           \amc_num_sign_int}
2217         \int_compare:nNnTF \amc_num_expo_int > 0 {
2218           \vspace{\AMCnumeric@Vspace}
2219           \vbox{\hbox{\AMCexponent}}
2220           \vspace{\AMCnumeric@Vspace}
2221           \vbox{\amc_num_integer_boxes:NnnNN
2222             \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
2223             \amc_num_expo_sign_int}
2224           } {}
2225         }}
2226       } {
2227         \amc_num_integer_boxes:NnnNN
2228         \amc_num_digits_clist { digit } \amc_num_decd_int \amc_num_sign_bool
2229         \amc_num_sign_int
2230         \int_compare:nNnTF \amc_num_expo_int > 0 {
2231           \hspace{\AMCnumeric@Hspace}\AMCexponent\hspace{\AMCnumeric@Hspace}
2232           \amc_num_integer_boxes:NnnNN
2233           \amc_num_expo_digits_clist { expo } { 0 } \amc_num_exposign_bool
2234           \amc_num_expo_sign_int
2235         } {}
2236       }
2237     }
2238   }
2239 \fi

```

And tell AMC that we finished with this question:

```

2240 \ifAMC@ensemble\else\vspace{\AMCpostNquest}\par\fi
2241 \ifAMC@ensemble\ifAMCformulaire@dedans
2242   \AMCmessage{FQ}
2243 \fi\fi
2244 }
2245 }
2246
2247 \cs_new_eq:NN \AMCnumericShow \amc_numeric_show:nn
2248

```

`\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.

```

2249 \cs_new:Npn \amc_numeric_hide:nn #1 #2 {
2250   \keys_set:nn { amcnumeric } { #2 }
2251   \AMCncontextGoto
2252   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi
2253 }
2254
2255 \cs_new_eq:NN \AMCnumericHide \amc_numeric_hide:nn
2256
2257 \ExplSyntaxOff
2258 \def\AMCnumericChoicesPlain{%
2259   \AMC@if@separate@question{\AMC@mem@category{numeric}}{%
2260     \AMCformatChoices{\AMCnumericShow}{\AMCnumericHide}%
2261   }

```

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...

```

2262 \ExplSyntaxOn
2263
2264 \cs_new:Npn \amc_numeric_choices:nn #1#2 {
2265   \AMCnumericChoicesPlain{#1}{#2}
2266 }
2267 \cs_generate_variant:Nn \amc_numeric_choices:nn { xn }
2268 \cs_new_eq:NN \AMCnumericChoices \amc_numeric_choices:xn
2269
2270 \ExplSyntaxOff

```

4.13.3 Intervals

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

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

As an example,

```

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

```
2271 \def\AMCIntervalFormat#1#2{[#1,\,#2]}
```

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

```
2272 \ExplSyntaxOn
```

```
2273
```

```
2274 \fp_new:N \amc_interv_a
```

```
2275 \fp_new:N \amc_interv_b
```

```
2276 \cs_new:Npn \amc_intervals:nnnn #1 #2 #3 #4 {
```

```
2277   \fp_set:Nn \amc_interv_a { #2 }
```

```
2278   \fp_do_while:nn { \amc_interv_a < #3 } {
```

```
2279     \fp_set:Nn \amc_interv_b { \amc_interv_a + #4 }
```

```
2280     \fp_compare:nTF { \amc_interv_a <= #1 < \amc_interv_b }
```

```
2281       \correctchoice \wrongchoice
```

```
2282       {\AMCIntervalFormat{\fp_use:N \amc_interv_a}{\fp_use:N \amc_interv_b}}
```

```
2283       \fp_set:Nn \amc_interv_a \amc_interv_b
```

```
2284   }
```

```
2285 }
```

```
2286 \cs_new_eq:NN \AMCIntervals \amc_intervals:nnnn
```

```
2287
```

```
2288 \ExplSyntaxOff
```

4.14 Open questions

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

As an example,

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

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

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

```
\end{question}
```

shows:

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

☐ w ☐ c

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

Begin with the options definitions:

```

2289 \def\AMCotextGoto{}
2290 \def\AMCotextReserved{}
2291 \def\AMCocol@Background{lightgray}
2292 \def\AMCocol@BoxFrameRule{white}
2293 \def\AMCocol@FrameRule{black}
2294 \def\AMCocol@Foreground{}
2295 \def\AMCopen@answer{}
2296 \def\AMCopen@question{}
2297 \def\AMCopen@lineuptext{}
2298 \define@key{AMCopen}{backgroundcol}{\def\AMCocol@Background{#1}}
2299 \define@key{AMCopen}{foregroundcol}{\def\AMCocol@Foreground{#1}}
2300 \define@key{AMCopen}{Treserved}{\def\AMCotextReserved{#1}}
2301 \define@key{AMCopen}{question}{\AMCid@name}{\def\AMCopen@question{#1}}
2302 \define@key{AMCopen}{answer}{\def\AMCopen@answer{#1}}
2303 \define@key{AMCopen}{contentcommand}{\AMCopen@lines}{\def\AMCopen@contentcommand{#1}}
2304 \newdimen\AMCopen@Hspace\AMCopen@Hspace=.5em
2305 \define@key{AMCopen}{hspace}{\AMCopen@Hspace=#1}
2306 \def\AMCopen@Width{.95\linewidth}
2307 \define@key{AMCopen}{width}{\def\AMCopen@Width{#1}}
2308 \newdimen\AMCopen@LineHeight\AMCopen@LineHeight=1cm
2309 \define@key{AMCopen}{lineheight}{\AMCopen@LineHeight=#1}
2310 \newcount\AMCopen@Lines\AMCopen@Lines=1
2311 \define@key{AMCopen}{lines}{\AMCopen@Lines=#1}
2312 \newdimen\AMCopen@boxmargin\AMCopen@boxmargin=3pt
2313 \define@key{AMCopen}{boxmargin}{\AMCopen@boxmargin=#1}
2314 \newdimen\AMCopen@boxframerule\AMCopen@boxframerule=1pt
2315 \define@key{AMCopen}{boxframerule}{\AMCopen@boxframerule=#1}
2316 \define@key{AMCopen}{boxframerulecol}{\def\AMCocol@BoxFrameRule{#1}}
2317 \define@key{AMCopen}{framerulecol}{\def\AMCocol@FrameRule{#1}}
2318 \newdimen\AMCopen@framerule\AMCopen@framerule=1pt
2319 \define@key{AMCopen}{framerule}{\AMCopen@framerule=#1}
2320 \define@key{AMCopen}{lineuptext}{\def\AMCopen@lineuptext{#1}}
2321 \define@boolkey{AMCopen}{dots}[true]{}
2322 \define@boolkey{AMCopen}{scan}[true]{}
2323 \define@boolkey{AMCopen}{retick}[true]{}
2324 \define@boolkey{AMCopen}{annotate}[false]{}
2325 \define@boolkey{AMCopen}{lineup}[false]{}
2326 \setkeys{AMCopen}{dots,scan,retick,annotate,lineup,contentcommand}
2327 \newcommand\AMCopenOpts[1]{\setkeys{AMCopen}{#1}}

```

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

```

2328 \newcommand\AMCopen@lines{%
2329   \begin{minipage}{\AMCopen@Width}%
2330     \loop\vspace{\AMCopen@LineHeight}
2331     \hspace*{.5em}\ifAMC@correc\smash{\AMCopen@answer}\def\AMCopen@answer{}\fi%
2332     \ifKV@AMCopen@dots%
2333     \dotfill\hspace*{.5em}
2334     \fi
2335     \ifnum\AMCopen@Lines>\@ne\par\advance\AMCopen@Lines\m@ne\repeat%
2336   \end{minipage}
2337 }
2338 \newcommand\AMCopenShow[2]{
2339   \ifAMC@ensemble\ifAMC@formulaire@dedans%
2340     \AMCmessage{Q=\the\AMCid@quest}%
2341     \fi\fi%
2342   {\setkeys{AMCopen}{#1}%
2343     \ifKV@AMCopen@lineup%
2344       \ifAMC@ensemble\else%
2345         \ifx\@empty\AMCopen@lineuptext\@empty\fi%
2346       \fi%
2347       \ifAMC@correc\smash{\AMCopen@answer}\fi%
2348       \ifx\@empty\AMCopen@lineuptext\@empty%
2349         \dotfill%
2350       \else%
2351         \AMCopen@lineuptext\hfill%
2352       \fi%
2353     \else%
2354       \hspace*{.5em}\linebreak[1]\hspace*{\fill}%
2355     \fi%
2356     {\AMCnoCompleteMulti%
2357       \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2358       \def\AMCanswer##1##2{\ifAMC@ensemble ##1\else%
2359         \ifAMC@inside@box ##1\else\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2360       \hspace{\AMCopen@Hspace}}%
2361       \fboxsep=\AMCopen@boxmargin%
2362       \fboxrule=\AMCopen@boxframerule%
2363       \fcolorbox{\AMCocol@BoxFrameRule}{\AMCocol@Background}{%
2364         \ifAMC@ensemble\AMCopen@question%
2365           \ifx\@empty\AMCopen@question\@empty\else\hspace{\AMCopen@Hspace}\fi%
2366         \fi%
2367         \begin{choicescustom}[o]%
2368           \ifx\AMCocol@Foreground\@empty\@empty\else%
2369             \def\AMC@boxcolor{\AMCocol@Foreground}%
2370           \fi%
2371           #2%
2372           \ifKV@AMCopen@scan\else\AMCdontScan\fi%
2373           \ifKV@AMCopen@retick\AMCreTick\fi%
2374           \ifKV@AMCopen@annotate\else\AMCdontAnnotate\fi%
2375         \end{choicescustom}%
2376         \ifx\@empty\AMCotextReserved\@empty%
2377           \hspace{-\AMCopen@Hspace}%
2378         \else%
2379           \ifx\AMCocol@Foreground\@empty\@empty%
2380             \AMCotextReserved%

```

```

2381         \else%
2382         \textcolor{\AMCocol@Foreground}{\AMCotextReserved}%
2383         \fi%
2384     \fi%
2385 }}%
2386 \ifKV@AMCOpen@lineup\else%
2387     \par\nobreak\noindent%
2388     \hspace*{\fill}{%
2389         \fboxrule=\AMCopen@framerule%
2390         \fcolorbox{\AMCocol@FrameRule}{white}{%
2391             \csname\AMCopen@contentcommand\endcsname
2392         }%
2393     \vspace{\AMCpostOquest}\par%
2394 \fi%
2395 }%
2396 \ifAMC@ensemble\ifAMCformulaire@dedans%
2397 \AMCmessage{FQ}%
2398 \fi\fi%
2399 }
2400 \newcommand\AMCopenHide[2]{%
2401     \AMCotextGoto%
2402     \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2403 }
2404 \def\AMCOpen{%
2405     \AMC@if@separate@question{\AMC@mem@category{open}}%
2406     \AMCformatChoices{\AMCopenShow}{\AMCopenHide}%
2407 }
2408 \ExplSyntaxOn
2409 \cs_new:Npn \amc_open_question:nn #1#2 {
2410     \AMCOpen{#1}{#2}
2411 }
2412 \cs_generate_variant:Nn \amc_open_question:nn { xn }
2413 \cs_new_eq:NN \AMCOpenX \amc_open_question:xn
2414 \ExplSyntaxOff

```

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}

2415 \def\AMCbotextGoto{}
2416 \def\AMCbo@help{}
2417 \define@key{AMCBoxOnly}{help}{\def\AMCbo@help{#1}}
2418 \define@boolkey{AMCBoxOnly}{ordered}[false]{}
2419 \setkeys{AMCBoxOnly}{ordered}

```

```

2420 \newcommand\AMCboOpts[1]{\setkeys{AMCBoxOnly}{#1}}
2421 \newcommand\AMCboShow[2]{%
2422   \ifAMC@ensemble\ifAMCformulaire@dedans%
2423     \AMCmessage{Q=\the\AMCid@quest}%
2424   \fi\fi%
2425   {\setkeys{AMCBoxOnly}{#1}%
2426     \def\AMCbeginAnswer{}\def\AMCendAnswer{}%
2427     \def\AMCanswer##1##2{\hspace{\AMCformHSpace} \ifAMC@ensemble ##1\else%
2428       \ifAMC@inside@box ##1\else{\AMCboxOutsideLetter{##1}{##2}}\fi\fi%
2429     }%
2430     \ifAMC@ensemble\AMCbo@help\fi%
2431     \ifKV@AMCBoxOnly@ordered%
2432       \begin{choicescustom}[o]%
2433     \else%
2434       \begin{choicescustom}%
2435     \fi%
2436     #2
2437   \end{choicescustom}%
2438 }%
2439 \ifAMC@ensemble\ifAMCformulaire@dedans%
2440 \AMCmessage{FQ}%
2441 \fi\fi%
2442 }
2443 \newcommand\AMCboHide[2]{
2444   \AMCbotextGoto%
2445   \ifAMC@qbloc\else\vspace{1.5ex}\par\fi%
2446 }
2447 \def\AMCBoxOnly{%
2448   \AMC@if@separate@question{\AMC@mem@category{box}}%
2449   \AMCformatChoices{\AMCboShow}{\AMCboHide}%
2450 }

```

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.

```

2451 \DeclareFontShape{OT1}{cmr}{b}{n}{<35->cmr17}{-}
2452 \def\AMC@watertext{\AMC@loc@draft}
2453 \newcommand\AMCw@termark{%
2454   \setlength{\@tempdimb}{.5\paperwidth}%
2455   \setlength{\@tempdimc}{-.5\paperheight}%
2456   \put(\strip@pt\@tempdimb,\strip@pt\@tempdimc){%
2457     \makebox(0,0){\rotatebox{45}{\AMC@LR{%
2458       \textcolor{gray}{0.8}{
2459         \fontencoding{OT1}\fontfamily{cmr}
2460         \fontseries{b}\fontshape{n}
2461         \fontsize{90pt}{120pt}
2462         \selectfont
2463         \AMC@watertext}}}}}}
2464 \newcommand\AMCw@terprint[1]{%
2465   \setbox\@tempboxa\vbox to \z@{

```

```

2466 \vbox{%
2467 \hbox to \z@{%
2468 #1\hss}}\vss}
2469 \dp\@tempboxa\z@
2470 \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}.

```

2471 \def\AMCcircle#1#2{%
2472 {\setlength{\unitlength}{1mm}%
2473 \begin{picture}(\#1,\#1)(-#2,-#2)\thinlines\circle*{\#1}\end{picture}}
2474 \def\m@rqueCalage{\AMCcircle{3.6}{1.8}}
2475 \def\m@rque#1{\AMC@tracebox{1}{\#1}{\m@rqueCalage}}
2476 \def\he@dtaille#1{% \par cancels the \leavevmode
2477 % introduced by https://github.com/pietvo/fancyhdr/commit/6b1ad10eeb5bc3d804f3cd2cf193e6440d0229e6
2478 \par\vbox to 1cm{\#1}}
2479 \def\he@dbas#1{\he@dtaille{\vspace*{\fill}\#1}}
2480 \def\he@dhaut#1{\he@dtaille{\#1\vspace*{\fill}}}
2481 \def\AMC@intituleHead{\AMC@loc@corrected}
2482 \def\AMC@note{}
2483 \def\AMCsetFoot#1{\def\AMC@note{\#1}}
2484 \newcommand\AMCStudentNumber{\the\AMCid@etud}
2485 \newcommand\AMCIDBoxesA{\AMC@binaryCode{id=1,ndigits=\AMC@NCBetud}{\the\AMCid@etud}}
2486 \newcommand\AMCIDBoxesB{\AMC@binaryCode{id=2,ndigits=\AMC@NCBpage}{\thepage}}
2487 \newcommand\AMCIDBoxesC{\AMC@binaryCode{id=3,ndigits=\AMC@NCBcheck}{\the\AMCid@check}}
2488 \newcommand\AMCIDBoxesABC{%
2489 \hbox{\vbox{\noindent\AMCIDBoxesA\
2490 \noindent\AMCIDBoxesB\AMCIDBoxesC}}}%
2491 }
2492 \def\AMC@pageHook{%
2493 \AMC@pagepos%
2494 \ifAMC@pagelayout\global\advance\AMCid@check\m@ne%
2495 \ifnum\AMCid@check<1\global\AMCid@check=\AMCid@checkmax\fi%
2496 \ifAMC@watermark\ifAMC@correthead\else\AMC@waterprint{\AMC@watermark}%
2497 \fi\fi\fi}
2498 \@ifl@t@r{\fmtversion{2020/10/01}}
2499 {\AddToHook{shipout/background}{\put(0in,0in){\AMC@pageHook}}}
2500 {\AtBeginShipout{\AtBeginShipoutUpperLeft{\AMC@pageHook}}}
2501 \fancypagestyle{AMCpageHeadOnly}{%
2502 \fancyhf{}\fancyhead[C]{\textsc{\AMC@intituleHead}}%
2503 \renewcommand{\headrulewidth}{0pt}%
2504 \renewcommand{\footrulewidth}{0pt}%
2505 }
2506 \fancypagestyle{AMCpageFull}{%
2507 \fancyhf}%

```

```

2508 \fancyhead[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHG}}}}%
2509 \fancyhead[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionHD}}}}%
2510 \fancyfoot[L]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionBG}}}}%
2511 \fancyfoot[R]{\AMC@LR{\he@dbas{\leavevmode\m@rque{positionBD}}}}%
2512 \fancyhead[C]{\AMC@LR{\he@dhaut{%
2513     \begin{minipage}[b]{\AMC@CBtaille}\AMCboxColor{black}%
2514     \ifAMCids@top\vbox to \AMCids@height{\texttt{+}\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2515     \AMCIDBoxesABC
2516     \end{minipage}%
2517     \ifAMCids@side\hbox to \AMCids@width{\hspace*{\fill}%
2518     \texttt{+}\the\AMCid@etud/\thepage/\the\AMCid@check+}}\fi%
2519 }}}%
2520 \fancyhfoffset[EOLR]{5mm}%
2521 \fancyfoot[C]{\AMC@note}%
2522 \renewcommand{\headrulewidth}{0pt}%
2523 \renewcommand{\footrulewidth}{0pt}%
2524 }
2525 \newcommand\AMCsubjectPageTag{%
2526     \fbox{\texttt{\the\AMCid@etud:\thepage}}%
2527 }
2528 \fancypagestyle{AMCpageNoMarks}{%
2529     \fancyhf{}%
2530     \fancyhead[R]{\AMCsubjectPageTag}%
2531     \fancyfoot[C]{\AMC@note}%
2532     \renewcommand{\headrulewidth}{0pt}%
2533     \renewcommand{\footrulewidth}{0pt}%
2534 }
2535 \fancypagestyle{AMCpageEmpty}{%
2536     \fancyhf{}%
2537     \renewcommand{\headrulewidth}{0pt}%
2538     \renewcommand{\footrulewidth}{0pt}%
2539 }
2540 \AtBeginDocument{%
2541     \ifAMC@pagelayout%
2542         \ifAMC@correthead
2543             \pagestyle{AMCpageHeadOnly}
2544         \else
2545             \pagestyle{AMCpageFull}
2546         \fi
2547     \fi
2548 }

```

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.

```

2549 \newcommand{\onecopy}[2]{%
2550     \ifx\AMCNombreCopies\undefined\AMCnum@copies=#1%
2551     \else\AMCnum@copies=\AMCNombreCopies\fi%
2552     \AMCmessage{TOTAL=\the\AMCnum@copies}%
2553     \message{^^JAMC:copies:total=\the\AMCnum@copies^^J}%
2554     \AMCid@etud=\AMCid@etudstart%

```

```

2555 \ifnum\AMCid@etud=0\AMCid@etud=\AMC@premierecopie\fi%
2556 \AMCid@etudfin=\AMCnum@copies%
2557 \advance\AMCid@etudfin\AMCid@etud\relax%
2558 \ifAMC@correthead\AMCid@etudfin=\AMC@premierecopie%
2559 \message{^^JAMC:copies:total=1^^J}%
2560 \fi
2561 \ifAMC@pdfform\begin{Form}\fi%
2562 \loop{%
2563 \global\AMCrep@nnmax=0%
2564 \ifAMC@calibration\protected@write\AMC@XYFILE{}\fi%
2565 \string\rngstate{\the\AMCid@etud}{\the\AMC@SR}%
2566 }\fi%
2567 \AMC@zoneformulairefalse\setcounter{page}{1}\setcounter{section}{0}%
2568 \ifAMC@ensemble\ifAMC@autemarks\pagestyle{AMCpageNoMarks}\fi\fi%
2569 \AMCnumero{1}%
2570 \ifAMC@calibration\AMCmessage{ETU=\the\AMCid@etud}\fi%
2571 \AMC@multiclear%
2572 \global\AMC@keepmemoryfalse%
2573 #2%
2574 \ifAMC@keepmemory\else\AMC@mem@clear\fi%
2575 \clearpage}%
2576 \message{^^JAMC:copies:add=1^^J}%
2577 \advance\AMCid@etud\@one\ifnum\AMCid@etud<\AMCid@etudfin\repeat%
2578 \global\AMCid@etudstart=\AMCid@etud%
2579 \ifAMC@pdfform\end{Form}\fi%
2580 \AMC@multi@report%
2581 }

```

\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.

```

2582 \newcount\AMC@addpages
2583 \newcommand{\AMCaddpagesto}[1]{%
2584 \AMC@addpages=#1\advance\AMC@addpages\@one%
2585 \clearpage%
2586 \@whilenum\thepage<\AMC@addpages\do{%
2587 \ifAMC@autemarks\pagestyle{AMCpageEmpty}\fi%
2588 \hbox{}\clearpage%
2589 }%
2590 }

```

\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).

```

2591 \def\AMCcleardoublepage{%
2592 \clearpage%
2593 \ifodd\thepage\else%
2594 \ifAMC@autemarks\pagestyle{AMCpageEmpty}\fi%
2595 \hbox{}\clearpage%
2596 \fi%
2597 }

```

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

```
2598 \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
2599 \newcommand\AMCstudentlabel[1]{\the\AMCid@etud-#1}
2600 \def\AMClabel#1{\expandafter\label{\AMCstudentlabel{#1}}}
2601 \def\AMCref#1{\expandafter\ref{\AMCstudentlabel{#1}}}
2602 \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.

```
2603 \newcommand{\AMCqlabel}[1]{%
2604   \AMClabel{#1}%
2605 }
```

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.

```
2606 \newcommand{\AMCassociation}[2][]{%
2607   \ifAMC@calibration%
2608     \immediate\write\AMC@XYFILE{\string\association{\the\AMCid@etud}{#2}{#1}}%
2609   \fi%
2610 }
```

`MCstudentslistfile` 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>}`.

```
2611 \newcommand{\AMCstudentslistfile}[2]{%
2612   \ifAMC@calibration%
2613     \immediate\write\AMC@XYFILE{\string\with{studentslistfile=#1}}%
2614     \immediate\write\AMC@XYFILE{\string\with{studentslistkey=#2}}%
2615   \fi%
2616 }
```

4.19 Package options

See section 3.1 for the options descriptions.

```
2617 \def\AMC@lang@code{}
2618 \DeclareOptionX{noshuffle}{\AMC@ordretrue}
2619 \DeclareOptionX{noshufflegroups}{\AMC@shuffleGfalse}
2620 \DeclareOptionX{fullgroups}{\AMC@fullGroupstrue}
2621 \DeclareOptionX{answers}{\AMC@corretheadtrue\AMC@correcttrue}
2622 \DeclareOptionX{indivanswers}{\AMC@correcttrue}
2623 \DeclareOptionX{textpos}{\AMC@textPostrue}
2624 \DeclareOptionX{extractonly}{\AMC@extractOnlytrue\AMC@textPostrue\AMC@boxStyle{shape=none}\AMC@boxedAnswer}
2625 \DeclareOptionX{box}{\AMC@qbloctrue}
2626 \DeclareOptionX{asbox}{\AMC@asqbloctrue}
2627 \DeclareOptionX{separateanswersheet}{\AMC@ensembletrue}
```

```

2628 \DeclareOptionX{digits}{\AMC@inside@digittrue}
2629 \DeclareOptionX{ordre}{\AMC@ordrettrue}
2630 \DeclareOptionX{correc}{\AMC@correcheadtrue\AMC@correcttrue}
2631 \DeclareOptionX{modele}{\AMC@correcheadtrue\AMC@correcfalse\AMC@ordrettrue}
2632 \DeclareOptionX{correcindiv}{\AMC@correcttrue}
2633 \DeclareOptionX{init}{\ifx\NoWatermarkExterne\undefined\relax\AMC@SR@time\fi}
2634 \DeclareOptionX{bloc}{\AMC@qbloctrue}
2635 \DeclareOptionX{completemulti}{\AMC@complete@multitrue}
2636 \DeclareOptionX{insidebox}{\AMC@inside@boxtrue}
2637 \DeclareOptionX{ensemble}{\AMC@ensembletrue}
2638 \DeclareOptionX{chiffres}{\AMC@inside@digittrue}
2639 \DeclareOptionX{outsidebox}{\AMC@outside@boxtrue}
2640 \DeclareOptionX{calibration}{\AMC@calibrationtrue}
2641 \DeclareOptionX{nowatermark}{\AMC@watermarkfalse}
2642 \newcommand\AMC@keys@next{\AMC@keys@linefalse}
2643 \newcommand\AMC@keys@line{\AMC@keys@linetrue}
2644 \DeclareOptionX{catalog}{\AMC@catalogtrue}
2645 \DeclareOptionX{flat}{\AMC@flattrue}
2646 \DeclareOptionX{keys}[next]{\csname AMC@keys@#1\endcsname{}}
2647 \DeclareOptionX{francais}{\def\AMC@lang@code{FR}\AMC@loc@FR}
2648 \DeclareOptionX{lang}{\def\AMC@lang@code{#1}\csname AMC@loc@#1\endcsname}
2649 \DeclareOptionX{versionA}{%
2650   \def\AMCid@checkmax{31}\def\AMC@NCBetud{9}\def\AMC@NCBpage{4}%
2651   \def\AMC@NCBcheck{5}\setlength{\AMC@CBtaille}{4cm}%
2652   \def\AMC@premierecopie{100}}
2653 \DeclareOptionX{plain}{\AMC@plaintrue}
2654 \DeclareOptionX{nopage}{\AMC@pagelayoutfalse}
2655 \DeclareOptionX{postcorrect}{\AMC@postcorrecttrue}
2656 \DeclareOptionX{automarks}{\AMC@automarkstrue}
2657 \newif\ifAMCneeds@storebox\AMCneeds@storeboxfalse
2658 \DeclareOptionX{storebox}{\AMCneeds@storeboxtrue}
2659 \DeclareOptionX{pdfform}{\AMC@pdfformtrue}
2660 \DeclareOptionX{codedigit}{\AMC@codeID@{#1}}
2661 \newif\ifAMC@survey\AMC@surveyfalse
2662 \DeclareOptionX{survey}{\AMC@surveytrue}
2663 \ifx\AMCPackageOptionsExterne\undefined\else%
2664   \expandafter\ExecuteOptionsX\expandafter{\AMCPackageOptionsExterne}\fi
2665 \ProcessOptionsX
2666 \ifAMCneeds@storebox
2667   \RequirePackage{storebox}\AtBeginDocument{{}}%
2668 \fi
2669 \ifAMC@pdfform
2670   \AMCmessage{VAR:project:pdfform=1}%
2671   \AMCboxStyle{shape=form}%
2672   \RequirePackage{pageanchor=false}{hyperref}%
2673 \else%
2674   \AMCmessage{VAR:project:pdfform=0}%
2675 \fi
2676 \AtBeginDocument{%
2677   \ifAMCneeds@storebox%
2678     \let\AMC@new@savebox=\newstorebox%
2679     \let\AMC@save@box=\storebox%
2680     \let\AMC@use@box=\usestorebox%

```

```

2681 \fi%
2682 \AMC@new@savebox{\AMC@ovalbox@R}%
2683 \AMC@new@savebox{\AMC@ovalbox@RF}%
2684 \AMC@new@savebox{\AMC@ovalbox@}%
2685 \AMC@new@savebox{\AMC@ovalbox@F}%
2686 \AMC@shapeprepare%
2687 }

```

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.) 2688 \ifAMC@survey
question-auto (env.) 2689 \NewEnviron{Questionnaires}[1]{
  values (env.) 2690 \onecopy{#1}{
values-auto (env.) 2691 \BODY
variable-auto (env.) 2692 }
  \answer 2693 }
    2694 \NewEnviron{auto}[1]{
    2695 \csvreader[head to column names, separator=tab]{#1}{}{
    2696 \BODY
    2697 }
    2698 }
    2699 \NewEnviron{question-auto}[3]{
    2700 \csvreader[head to column names, separator=tab]{#1}{}{
    2701 \ifcsvstrcmp{#2}{#3}{\BODY \\\}
    2702 }
    2703 }
    2704 \newenvironment{values}{}{}
    2705 \NewEnviron{values-auto}[5]{
    2706 \csvreader[head to column names, separator=tab]{#1}{}{
    2707 \ifcsvstrcmp{#2}{#3}{
    2708 \ifcsvstrcmp{#4}{#5}{\BODY \\\}
    2709 }{}
    2710 }{}
    2711 }
    2712 }
    2713 \NewEnviron{variable-auto}[3]{
    2714 \foreach \x in {#3}{
    2715 \csvreader[head to column names, separator=tab]{#1}{}{
    2716 \ifcsvstrcmp{#2}{\x}{\BODY}{
    2717 }
    2718 }
    2719 }
    2720 \newcommand{\answer}[5][\global\advance\AMCrep@count\@ne\relax%
    2721 \ifAMC@calibration\AMCmessage{REP=\the\AMCrep@count:B}\fi%
    2722 \global\AMC@bonnettrue%
    2723 \AMCload@reponse{\une@rep{\ifAMC@correc\AMC@box{#1}{\AMC@checkedbox}%
    2724 \else\AMC@box{#1}{} \fi}{#2}{#3}{#4}{#5}}{\the\AMCrep@count}\ignorespaces}
    2725 \RequirePackage{tikz}

```

```

2726 \usetikzlibrary{positioning, shapes, arrows, tikzmark, decorations.pathreplacing}
2727 \tikzset{
2728     checkbox-sc/.style={
2729         right=of lab\thecsvrow
2730     },
2731     vallab-sc/.style={
2732         text width=4cm,
2733         align=left,
2734     },
2735     checkbox-mc/.style={
2736     },
2737     vallab-mc/.style={
2738         above=of box\thecsvrow,
2739         text width=1.4cm,
2740         align=center,
2741     },
2742     varlab-mc/.style={
2743         text width=4cm,
2744         align=left,
2745     },
2746     node distance= 0mm,
2747 }
2748 \fi

```

4.21 Package Errors

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

```

2749 \def\AMC@error@explain{\PackageError{automultiplechoice}{
2750     Command \protect\explain\space can only be used inside\MessageBreak question like environments}{Some
2751 }}

```

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.

```

2752 \ifAMC@plain
2753 \else
2754     \IfFileExists{environ.sty}{\RequirePackage{environ}}{}
2755     \ifx\TeXversion\@undefined
2756     \else
2757         \RequirePackage{etex}
2758     \fi
2759 \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 $\text{\LaTeX}$ code which makes *one* exam copy. Optional argument $\langle n \rangle$ gives the number of desired copies – this can also be modified redefinig `\AMCNombreCopies`.

```

2760 \@ifpackageloaded{environ}{%
2761     \ifx\NewEnviron\undefined\PackageWarning{automultiplechoice}%
2762     {Package environ loaded but too old version:

```

```

2763     environnement examcopy/copieexamen will NOT be defined.}%
2764     \else\NewEnviron{examcopy}[1][5]{\onecopy{#1}{\BODY}}\fi}%
2765 {\PackageWarning{automultiplechoice}%
2766   {Package environ not loaded: environnement
2767     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.

```

2768 \ExplSyntaxOn
2769
2770 \cs_if_exist:NTF \pdfsavepos { } {
2771   \cs_if_exist:NTF \savepos { \cs_new_eq:NN \pdfsavepos \savepos } { }
2772 }
2773 \cs_if_exist:NTF \pdflastxpos { } {
2774   \cs_if_exist:NTF \lastxpos { \cs_new_eq:NN \pdflastxpos \lastxpos } { }
2775 }
2776 \cs_if_exist:NTF \pdflastypos { } {
2777   \cs_if_exist:NTF \lastypos { \cs_new_eq:NN \pdflastypos \lastypos } { }
2778 }

```

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

```

2779
2780 \cs_if_exist:NTF \pdfpagewidth { } {
2781   \cs_new_eq:NN \pdfpagewidth \paperwidth
2782 }
2783 \cs_if_exist:NTF \pdfpageheight { } {
2784   \cs_new_eq:NN \pdfpageheight \paperheight
2785 }
2786
2787 \ExplSyntaxOff

```

4.24 External control

```

\SujetExterne Some of the package options can be controlled defining \xxxExterne commands. For example,
\ScoringExterne the following command will format the subject document, whatever options are used in the LATEX
\CorrigeExterne file:
\CorrigeIndivExterne pdflatex '\nonstopmode\def\SujetExterne{1}\def\NoWatermarkExterne{1}\input{mcq.tex}'
\NoWatermarkExterne
2788 \def\AMC@randomseeds#1{%
2789   \ifx\pgfmath@rnd@z\undefined\relax
2790     \AMCmessage{VAR:pgfseed:#1=NONE}%
2791   \else%
2792     \pgfmathparse{\pgfmath@rnd@z}%
2793     \AMCmessage{VAR:pgfseed:#1=\pgfmathresult}%
2794   \fi%
2795 }
2796 \def\AMC@randomseeds#1{%
2797   \AtBeginDocument{\AMC@randomseeds@{#1}}%

```

```

2798 }
2799 \ifx\SujetExterne\undefined\else
2800 \message{***SUJET***^^J}\AMC@randomseeds{subject}
2801 \AMC@catalogfalse\AMC@flatfalse\AMC@calibrationtrue\AMC@correcfalse\AMC@correheadfalse\AMC@watermarkfalse
2802 \fi
2803 \ifx\ScoringExterne\undefined\else
2804 \message{***SCORING***^^J}\AMC@randomseeds{scoring}
2805 \AMC@catalogfalse\AMC@flatfalse\AMC@calibrationtrue\AMC@correcfalse\AMC@correheadfalse\AMC@watermarkfalse
2806 \fi
2807 \ifx\CorrigeExterne\undefined\else
2808 \message{***CORRIGE***^^J}
2809 \AMC@catalogfalse\AMC@flatfalse\AMC@calibrationfalse\AMC@correheadtrue\AMC@correcttrue\AMC@watermarkfalse
2810 \fi
2811 \ifx\CorrigeIndivExterne\undefined\else
2812 \message{***CORRIGE***^^J}\AMC@randomseeds{indivanswer}
2813 \AMC@catalogfalse\AMC@flatfalse\AMC@calibrationtrue\AMC@correheadfalse\AMC@correcttrue\AMC@watermarkfalse
2814 \fi
2815 \ifx\CatalogExterne\undefined\else
2816 \message{***CATALOG***^^J}
2817 \AMC@catalogtrue\AMC@flatfalse
2818 \fi
2819 \ifx\NoWatermarkExterne\undefined\else
2820 \AMC@watermarkfalse
2821 \fi
2822 \ifx\codeDigitExterne\undefined\else
2823 \AMCcodeID@{\codeDigitExterne}
2824 \fi
2825 \ifAMC@catalog
2826 \AMC@watermarkfalse\AMC@correheadtrue%
2827 \AMC@correcttrue\AMC@ordretrue\AMC@shuffleGfalse%
2828 \AMC@fullGroupstrue%
2829 \def\AMC@intituleHead{\AMC@loc@catalog}\AMC@affichekeystrue
2830 \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:

```

2831 \@ifpackageloaded{geometry}{-}{\usepackage{geometry}}
2832 \ifAMC@pagelayout
2833 \ifAMC@correhead
2834 \geometry{hmargin=3cm,vmargin={1cm,1cm},includeheadfoot,headheight=1cm,footskip=1cm}
2835 \else
2836 \geometry{hmargin=3cm,headheight=2cm,headsep=.3cm,footskip=1cm,top=3.5cm,bottom=2.5cm}
2837 \fi
2838 \ifAMC@watermark
2839 \ifAMC@correhead\else
2840 \def\AMC@note{\begin{minipage}{0.65\linewidth}
2841 \AMC@LR{\textcolor{blue}{\AMC@loc@message}}
2842 \end{minipage}}
2843 }
2844 \fi
2845 \fi

```

2846 \fi

4.26 Initialisation

Initialisation of the check counter:

2847 \AMCid@check=\AMCid@checkmax

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

2848 \ifAMC@ensemble\AMCmessage{VAR:ensemble=1}\fi

2849 \ifAMC@inside@box\AMCmessage{VAR:insidebox=1}\fi

2850 \ifAMC@outside@box\AMCmessage{VAR:outsidebox=1}\fi

2851 \ifAMC@postcorrect\AMCmessage{VAR:postcorrect=1}\fi

Preparing writing to .xy file :

2852 \ifAMC@calibration

2853 \newwrite\AMC@XYFILE%

2854 \immediate\openout\AMC@XYFILE\jobname.xy%

2855 \immediate\write\AMC@XYFILE{\string\version{\AMC@VERSION}}

2856 \immediate\write\AMC@XYFILE{\string\with{codedigit=\AMCcodeID@mode}}

2857 \immediate\write\AMC@XYFILE{\string\with{version=\AMC@VERSION}}

2858 \immediate\write\AMC@XYFILE{\string\with{ensemble=\ifAMC@ensemble yes\else no\fi}}

2859 \immediate\write\AMC@XYFILE{\string\with{insidebox=\ifAMC@inside@box yes\else no\fi}}

2860 \immediate\write\AMC@XYFILE{\string\with{outsidebox=\ifAMC@outside@box yes\else no\fi}}

2861 \immediate\write\AMC@XYFILE{\string\with{postcorrect=\ifAMC@postcorrect yes\else no\fi}}

2862 \immediate\write\AMC@XYFILE{\string\with{extractonly=\ifAMC@extractOnly yes\else no\fi}}

2863 \immediate\write\AMC@XYFILE{\string\with{lang=\AMC@lang@code}}

2864 \ifx\AMCNombreCopies\undefined%

2865 \immediate\write\AMC@XYFILE{\string\with{ncopies=default}}}%

2866 \else%

2867 \immediate\write\AMC@XYFILE{\string\with{ncopies=\AMCNombreCopies}}}%

2868 \fi%

2869 \fi

4.27 French command names

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

2870 \let\reponses=\choices\let\endreponses=\endchoices

2871 \let\reponseshoriz=\choiceshoriz\let\endreponseshoriz=\endchoiceshoriz

2872 \let\reponsesperso=@choicescustom\let\endreponsesperso=\endchoicescustom

2873 \let\bonne=\correctchoice

2874 \let\mauvaise=\wrongchoice

2875 \let\bareme=\scoring

2876 \let\baremeDefautM=\scoringDefaultM

2877 \let\baremeDefautS=\scoringDefaultS

2878 \def\exemplaire{\AMC@loc@FR\onecopy}

2879 \@ifpackageloaded{environ}{%

2880 \let\copieexamen=examcopy\let\endcopieexamen=\endexamcopy}{}

2881 \let\melangegroupe=\shufflegroup

2882 \let\restituegroupe=\insertgroup

2883 \let\alafin=\lastchoices

2884 \let\formulaire=\AMCform

2885 \let\AMCdebutFormulaire=\AMCformBegin

2886 \let\champnom=\namefield

2887 \let\choixIntervalles=\AMCIntervals

4.28 flat mode

```

2888 \ifAMC@flat
2889   \renewcommand{\onecopy}[1]{}
2890   \renewcommand{\examcopy}[1]{}{}
2891   \renewcommand{\copieexamen}[1]{}{}
2892   \renewcommand{\endexamcopy}{}
2893   \renewcommand{\endcopieexamen}{}
2894   \renewcommand{\element}[1]{\noindent$\longrightarrow$ \texttt{#1}\nopagebreak\par}
2895   \renewcommand{\insertgroup}[2][0]{\par--- \texttt{#2} ---\par}
2896   \renewcommand{\copygroup}[3][0]{\par--- \texttt{#2}$\longrightarrow$\texttt{#2} ---\par}
2897   \renewcommand{\cleargroup}[1]{}
2898   \renewcommand{\insertgroupfrom}[3][0]{\par--- \texttt{#2}[#3,...] ---\par}
2899   \renewcommand{\copygroupfrom}[4][0]{\par--- \texttt{#2}[#4,...]$\longrightarrow$\texttt{#2} ---\par}
2900   \AMC@ordretrue
2901 \fi

```

5 Outputs

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

5.1 namefield command

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

```

\tracepos{0/35:__zone:id:__n}{0sp}{15446557sp}{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}{12461379sp}{square}

```

5.2 AMCboxedchar command

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

```

\tracepos{0/36:test}{22855914sp}{38025531sp}{square}
\tracepos{0/36:test}{23561334sp}{37320111sp}{square}

```

5.3 AMCcode command

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

```

\tracepos{0/61:case:code[5]:16,1}{24875505sp}{41571962sp}{square}
\tracepos{0/61:case:code[5]:16,1}{25580925sp}{40866542sp}{square}
\boxchar{0/61:case:code[5]:16,1}{A}
\tracepos{0/61:case:code[5]:16,2}{24875505sp}{40457850sp}{square}

```

```

\tracepos{0/61:case:code[5]:16,2}{25580925sp}{39752430sp}{square}
\boxchar{0/61:case:code[5]:16,2}{B}
\tracepos{0/61:case:code[5]:16,3}{24875505sp}{39343738sp}{square}
\tracepos{0/61:case:code[5]:16,3}{25580925sp}{38638318sp}{square}
\boxchar{0/61:case:code[5]:16,3}{C}
\tracepos{0/61:case:code[5]:16,4}{24875505sp}{38229626sp}{square}
\tracepos{0/61:case:code[5]:16,4}{25580925sp}{37524206sp}{square}
\boxchar{0/61:case:code[5]:16,4}{D}
\tracepos{0/61:case:code[4]:17,1}{26540304sp}{43800186sp}{square}
\tracepos{0/61:case:code[4]:17,1}{27245724sp}{43094766sp}{square}
\boxchar{0/61:case:code[4]:17,1}{0}
\tracepos{0/61:case:code[4]:17,2}{26540304sp}{42686074sp}{square}
\tracepos{0/61:case:code[4]:17,2}{27245724sp}{41980654sp}{square}
\boxchar{0/61:case:code[4]:17,2}{1}
\tracepos{0/61:case:code[4]:17,3}{26540304sp}{41571962sp}{square}
\tracepos{0/61:case:code[4]:17,3}{27245724sp}{40866542sp}{square}
\boxchar{0/61:case:code[4]:17,3}{2}
\tracepos{0/61:case:code[4]:17,4}{26540304sp}{40457850sp}{square}
\tracepos{0/61:case:code[4]:17,4}{27245724sp}{39752430sp}{square}
\boxchar{0/61:case:code[4]:17,4}{3}
\tracepos{0/61:case:code[4]:17,5}{26540304sp}{39343738sp}{square}
\tracepos{0/61:case:code[4]:17,5}{27245724sp}{38638318sp}{square}
\boxchar{0/61:case:code[4]:17,5}{4}
\tracepos{0/61:case:code[4]:17,6}{26540304sp}{38229626sp}{square}
\tracepos{0/61:case:code[4]:17,6}{27245724sp}{37524206sp}{square}
\boxchar{0/61:case:code[4]:17,6}{5}
\tracepos{0/61:case:code[3]:18,1}{28032161sp}{43800186sp}{square}
\tracepos{0/61:case:code[3]:18,1}{28737581sp}{43094766sp}{square}
\boxchar{0/61:case:code[3]:18,1}{0}
\tracepos{0/61:case:code[3]:18,2}{28032161sp}{42686074sp}{square}
\tracepos{0/61:case:code[3]:18,2}{28737581sp}{41980654sp}{square}
\boxchar{0/61:case:code[3]:18,2}{1}
\tracepos{0/61:case:code[3]:18,3}{28032161sp}{41571962sp}{square}
\tracepos{0/61:case:code[3]:18,3}{28737581sp}{40866542sp}{square}
\boxchar{0/61:case:code[3]:18,3}{2}
\tracepos{0/61:case:code[3]:18,4}{28032161sp}{40457850sp}{square}
\tracepos{0/61:case:code[3]:18,4}{28737581sp}{39752430sp}{square}
\boxchar{0/61:case:code[3]:18,4}{3}
\tracepos{0/61:case:code[3]:18,5}{28032161sp}{39343738sp}{square}
\tracepos{0/61:case:code[3]:18,5}{28737581sp}{38638318sp}{square}
\boxchar{0/61:case:code[3]:18,5}{4}
\tracepos{0/61:case:code[3]:18,6}{28032161sp}{38229626sp}{square}
\tracepos{0/61:case:code[3]:18,6}{28737581sp}{37524206sp}{square}
\boxchar{0/61:case:code[3]:18,6}{5}
\tracepos{0/61:case:code[2]:19,1}{29524018sp}{43800186sp}{square}
\tracepos{0/61:case:code[2]:19,1}{30229438sp}{43094766sp}{square}
\boxchar{0/61:case:code[2]:19,1}{0}
\tracepos{0/61:case:code[2]:19,2}{29524018sp}{42686074sp}{square}
\tracepos{0/61:case:code[2]:19,2}{30229438sp}{41980654sp}{square}

```

```

\boxchar{0/61:case:code[2]:19,2}{1}
\tracepos{0/61:case:code[2]:19,3}{29524018sp}{41571962sp}{square}
\tracepos{0/61:case:code[2]:19,3}{30229438sp}{40866542sp}{square}
\boxchar{0/61:case:code[2]:19,3}{2}
\tracepos{0/61:case:code[2]:19,4}{29524018sp}{40457850sp}{square}
\tracepos{0/61:case:code[2]:19,4}{30229438sp}{39752430sp}{square}
\boxchar{0/61:case:code[2]:19,4}{3}
\tracepos{0/61:case:code[2]:19,5}{29524018sp}{39343738sp}{square}
\tracepos{0/61:case:code[2]:19,5}{30229438sp}{38638318sp}{square}
\boxchar{0/61:case:code[2]:19,5}{4}
\tracepos{0/61:case:code[2]:19,6}{29524018sp}{38229626sp}{square}
\tracepos{0/61:case:code[2]:19,6}{30229438sp}{37524206sp}{square}
\boxchar{0/61:case:code[2]:19,6}{5}
\tracepos{0/61:case:code[1]:20,1}{31015875sp}{43800186sp}{square}
\tracepos{0/61:case:code[1]:20,1}{31721295sp}{43094766sp}{square}
\boxchar{0/61:case:code[1]:20,1}{0}
\tracepos{0/61:case:code[1]:20,2}{31015875sp}{42686074sp}{square}
\tracepos{0/61:case:code[1]:20,2}{31721295sp}{41980654sp}{square}
\boxchar{0/61:case:code[1]:20,2}{1}
\tracepos{0/61:case:code[1]:20,3}{31015875sp}{41571962sp}{square}
\tracepos{0/61:case:code[1]:20,3}{31721295sp}{40866542sp}{square}
\boxchar{0/61:case:code[1]:20,3}{2}
\tracepos{0/61:case:code[1]:20,4}{31015875sp}{40457850sp}{square}
\tracepos{0/61:case:code[1]:20,4}{31721295sp}{39752430sp}{square}
\boxchar{0/61:case:code[1]:20,4}{3}
\tracepos{0/61:case:code[1]:20,5}{31015875sp}{39343738sp}{square}
\tracepos{0/61:case:code[1]:20,5}{31721295sp}{38638318sp}{square}
\boxchar{0/61:case:code[1]:20,5}{4}
\tracepos{0/61:case:code[1]:20,6}{31015875sp}{38229626sp}{square}
\tracepos{0/61:case:code[1]:20,6}{31721295sp}{37524206sp}{square}
\boxchar{0/61: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	32
4.6.3	Tokens shuffling	32
4.7	Keys numbering	33
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	36

4.8.4	Scoring zones	41
4.8.5	Binary boxes	41
4.9	Checking Environment	43
4.10	Handling groups of questions	43
4.11	Questions	47
4.11.1	Managing answers	48
4.11.2	Separate answer sheet	49
4.11.3	Formatting answers	54
4.11.4	Score zones	56
4.11.5	Formatting questions	58
4.11.6	Explanations	60
4.12	Scoring	60
4.13	Numerical data	61
4.13.1	Codes	61
4.13.2	Numerical questions	64
4.13.3	Intervals	80
4.14	Open questions	81
4.15	Boxes with letters only	84
4.16	Page formatting	85
4.16.1	Watermark	85
4.16.2	Signs for scan analysis	86
4.17	Defining a single exam copy content	87
4.18	Pre-association	89
4.19	Package options	89
4.20	Survey add-on	91
4.21	Package Errors	92
4.22	Optional features	92
4.23	Use with recent LuaTeX versions	93
4.24	External control	93
4.25	Page layout	94
4.26	Initialisation	95
4.27	French command names	95
4.28	flat mode	96
5	Outputs	96
5.1	namefield command	96
5.2	AMCboxedchar command	96
5.3	AMCcode command	96

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				1494,	1496,	1499,	1884,	1885,	1887,
\"		151,	157	1500,	1501,	1502,	1889,	1892,	1897,
\@aucune		870,	874, 875	1504,	1506,	1509,	1898,	1900,	1904,
\@firstoftwo		688		1510,	1512,	1513,	1906,	1909,	1910,
\@ifl@t@r		3,	2498	1515,	1516,	1517,	1911,	1912,	1913,
\@ifstar		1107,	1108	1518,	1538,	1539,	1915,	1916,	1917,
\@secondoftwo		690		1544,	1545,	1546,	1921,	1922,	1923,
\@skiphyperreffalse	...	1289		1547,	1548,	1549,	1924,	1928,	1931,
\@skiphyperreftrue		1285,	1289	1554,	1556,	1557,	1936,	1938,	1941,
\@tempboxa	.	2465,	2469, 2470	1558,	1559,	1560,	1944,	1946,	1947,
\@tempdimb		2454,	2456	1562,	1564,	1565,	1948,	1952,	1953,
\@tempdimc		2455,	2456	1566,	1567,	1570,	1954,	1957,	1960,
_		1325		1577,	1578,	1579,	1964,	1968,	1970,
\~		192,	193, 198, 199	1581,	1583,	1584,	1972,	1975,	1980,
A				1586,	1588,	1590,	1983,	1984,	1992,
\aa		178		1593,	1594,	1596,	1998,	1999,	2006,
\AddToHook		2499		1598,	1599,	1600,	2007,	2009,	2010,
\alafin		21,	2883	1602,	1607,	1612,	2011,	2013,	2016,
\amc	234, 235, 237, 238, 239,			1624,	1662,	1664,	2017,	2020,	2021,
	240, 243, 246, 248,			1666,	1668,	1670,	2023,	2029,	2033,
	249, 250, 253, 255,			1672,	1675,	1678,	2034,	2036,	2039,
	897, 908, 910, 912,			1681,	1684,	1687,	2040,	2047,	2057,
	913, 915, 918, 919,			1690,	1693,	1696,	2058,	2059,	2060,
	921, 924, 925, 928,			1699,	1702,	1704,	2061,	2062,	2063,
	929, 930, 931, 933,			1706,	1709,	1718,	2064,	2065,	2066,
	935, 936, 938, 940,			1720,	1728,	1732,	2068,	2073,	2074,
	941, 943, 945, 946,			1733,	1734,	1740,	2077,	2083,	2085,
	949, 950, 952, 953,			1742,	1754,	1755,	2087,	2088,	2091,
	955, 956, 958, 959,			1756,	1757,	1758,	2092,	2096,	2097,
	961, 962, 964, 967,			1759,	1760,	1761,	2098,	2100,	2101,
	968, 970, 972, 976,			1764,	1767,	1771,	2103,	2104,	2105,
	977, 978, 979, 1390,			1774,	1775,	1779,	2106,	2107,	2109,
	1391, 1392, 1393,			1783,	1784,	1788,	2110,	2113,	2114,
	1394, 1395, 1396,			1792,	1793,	1797,	2116,	2117,	2120,
	1398, 1405, 1408,			1802,	1804,	1806,	2125,	2126,	2127,
	1413, 1417, 1425,			1813,	1815,	1818,	2128,	2130,	2131,
	1428, 1429, 1430,			1827,	1828,	1829,	2132,	2133,	2134,
	1433, 1438, 1440,			1830,	1831,	1832,	2135,	2136,	2139,
	1443, 1444, 1445,			1834,	1837,	1839,	2140,	2141,	2142,
	1446, 1448, 1450,			1840,	1841,	1842,	2146,	2148,	2150,
	1452, 1453, 1455,			1848,	1849,	1851,	2153,	2158,	2159,
	1456, 1460, 1461,			1852,	1853,	1854,	2160,	2161,	2162,
	1462, 1464, 1465,			1855,	1856,	1857,	2163,	2164,	2166,
	1466, 1468, 1473,			1859,	1860,	1861,	2167,	2168,	2169,
	1475, 1477, 1485,			1863,	1865,	1867,	2173,	2174,	2176,
	1490, 1491, 1492,			1870,	1872,	1874,	2182,	2189,	2194,
				1876,	1882,	1883,	2196,	2197,	2203,

2212, 2214, 2215,	\AMC@error@explain	\AMC@loc@NO 176
2216, 2217, 2221,	 1364, 1366, 2749	\AMC@loc@none . . 104, 117,
2222, 2223, 2227,	\AMC@fillcolor@	130, 144, 158, 171,
2228, 2229, 2230,	. 409, 427, 428, 444, 449	184, 197, 210, 223, 870
2232, 2233, 2234,	\AMC@fin@rep . . 857, 1119,	\AMC@loc@PT 189
2247, 2249, 2255,	1123, 1127, 1130, 1133	\AMC@loc@q 100, 113,
2264, 2267, 2268,	\AMC@formBox 549	127, 140, 154, 168,
2274, 2275, 2276,	\AMC@formBox@ 549	181, 193, 207, 219, 1292
2277, 2278, 2279,	\AMC@formcolspecs 1036, 1090	\AMC@loc@qf 99, 112,
2280, 2282, 2283,	\AMC@formspecs . . 1027, 1042	126, 139, 153, 167,
2286, 2409, 2412, 2413	\AMC@fullGroupsfalse . . . 36	180, 192, 206, 218, 879
\AMC@addpages 2582, 2584, 2586	\AMC@fullGroupstrue	\AMC@loc@question
\AMC@affiche 308, 1300	 2620, 2828	 105, 118,
\AMC@amclog 19	\AMC@if@separate@question	131, 145, 159, 172,
\AMC@answerBox 507	 908,	185, 198, 211, 224, 230
\AMC@answerBox@ . . . 388,	983, 988, 996, 1002,	\AMC@loc@questions
515, 562, 564, 580,	1523, 2259, 2405, 2448	 106, 119,
582, 648, 650, 658, 660	\AMC@imax . 742, 754, 755, 756	132, 146, 160, 173,
\AMC@binaryBoxes 681	\AMC@intituleHead	186, 199, 212, 225, 230
\AMC@binaryCode . . . 631,	 2481, 2502, 2829	\AMC@logchar 316, 417
682, 2485, 2486, 2487	\AMC@keepmemoryfalse . . 2572	\AMC@logfile . 19, 20, 21, 317
\AMC@box 549, 1169,	\AMC@keepmemorytrue . . 1024	\AMC@LR 23,
1170, 1174, 2723, 2724	\AMC@keyBox@	419, 2457, 2508, 2509,
\AMC@boxcolor 406, 532, 2369	. . 578, 1416, 1747, 1750	2510, 2511, 2512, 2841
\AMC@boxcolor@	\AMC@keys@line 2643	\AMC@makeovalbox
. 406, 407, 408, 410,	\AMC@keys@next 2642	. 442, 461, 462, 463, 464
428, 437, 449, 453, 470	\AMC@lang@code	\AMC@mem@add 938,
\AMC@boxeddown 420, 520, 531	. 2617, 2647, 2648, 2863	984, 991, 998, 1005, 1524
\AMC@boxedheight	\AMC@loc@CA 108	\AMC@mem@add@ifneeded
. 433, 447, 448, 450,	\AMC@loc@catalog 102, 115,	 895, 1332
451, 454, 455, 499,	129, 142, 156, 170,	\AMC@mem@addsingle@ifneeded
522, 526, 527, 539, 542	183, 195, 209, 221, 2829	 987,
\AMC@boxedrule 425,	\AMC@loc@corrected	1100, 1102, 1104, 1106
447, 448, 449, 518, 529	 101, 114,	\AMC@mem@addvar 943
\AMC@boxedwidth 436, 447,	128, 141, 155, 169,	\AMC@mem@aid 962, 997
448, 450, 451, 454,	182, 194, 208, 220, 2481	\AMC@mem@answer
455, 499, 521, 526,	\AMC@loc@DE 149	. . . 994, 1134, 1141,
528, 539, 540, 1070, 1905	\AMC@loc@draft . . 96, 109,	1142, 1143, 1148, 1154
\AMC@CBtaille 629, 2513, 2651	123, 136, 150, 164,	\AMC@mem@category . 956,
\AMC@checkbox	177, 190, 203, 216, 2452	990, 2259, 2405, 2448
. 388, 409, 412, 462,	\AMC@loc@ES 202	\AMC@mem@clear . . . 913, 2574
464, 474, 480, 490,	\AMC@loc@explain 103, 116,	\AMC@mem@next 933, 989, 1003
648, 650, 1169, 1745,	143, 157, 196, 222, 1364	\AMC@mem@openQuestion
1765, 1779, 1784, 2723	\AMC@loc@FR . . 135, 2647, 2878	 1001, 1331
\AMC@chiffres 1378	\AMC@loc@IT 163	\AMC@mem@qidaffname 950, 1004
\AMC@coli	\AMC@loc@JA 215	\AMC@mem@show 977, 1014, 1023
. 1035, 1039, 1040, 1041	\AMC@loc@message 97, 110,	\AMC@mem@show@filter
\AMC@crosschar 431, 534	124, 137, 151, 165,	 978, 1018
\AMC@crossrule 453, 519, 535	178, 191, 204, 217, 2841	\AMC@mn@leftmargin
\AMC@definitnumero . 309, 313	\AMC@loc@namesurname . . .	. 1180, 1188, 1191, 1197
\AMC@draw@crossfalse . . 411	. 107, 120, 133, 147,	\AMC@mn@rightmargin
\AMC@draw@crosstrue . . . 413	161, 174, 187, 200, 213	. 1181, 1189, 1192, 1195
	\AMC@loc@NL 122	\AMC@mn@sep . 1179, 1195, 1197

\AMC@mn@test	1178, 1185, 1194	\AMC@setcolors@		\AMC@sz@callin	
\AMC@multi@report	1443, 2580		405, 426, 443, 468	..	1213, 1218, 1226,
\AMC@multiclear	1448, 2571	\AMC@shape@form			1228, 1293, 1327, 1328
\AMC@NCBcheck	625, 2487, 2651	\AMC@shape@form@base	489, 503	\AMC@sz@callin@question	1203
\AMC@NCBetud	625, 2485, 2650	\AMC@shape@form@ticked		\AMC@sz@callout	
\AMC@NCBpage	625, 2486, 2650		491, 493, 496		1212, 1217,
\AMC@new@savebox		\AMC@shape@none			1221, 1223, 1320, 1321
.....	389, 2678,	\AMC@shape@oval		\AMC@sz@callout@margin	1205
	2682, 2683, 2684, 2685	\AMC@shape@square		\AMC@sz@callout@margin	1208
\AMC@note		\AMC@shapename		\AMC@sz@depth	1201, 1211, 1216
.....	2482,	..	322, 330, 339, 348, 525	\AMC@sz@height	
.....	2483, 2521, 2531, 2840	\AMC@shapename@	322, 420, 536		1201, 1210, 1215
\AMC@numeric@alsocorrect		\AMC@shapeprepare		\AMC@sz@init@margin	1207
.....	1724,		536, 544, 2686	\AMC@sz@width	1201, 1209, 1214
.....	2185, 2189, 2199, 2203	\AMC@shapeprepare@form	488	\AMC@sza@box	1236, 1244, 1247
\AMC@numeric@keepas		\AMC@shapeprepare@none	505	\AMC@sza@callin	889, 890,
.....	1722, 2178, 2179	\AMC@shapeprepare@oval	459		1258, 1263, 1271, 1273
\AMC@numeric@scoreapprox		\AMC@shapeprepare@square	423	\AMC@sza@callin@margin	1248
.....	1714, 2144, 2154	\AMC@shuffletoks		\AMC@sza@callin@margin	1252
\AMC@numeric@scoreexact			289, 775, 778, 850	\AMC@sza@callin@none	1240
.....	1712, 2143, 2149	\AMC@smashbox	..	\AMC@sza@callin@question	
\AMC@numeric@scorewrong		397, 398, 399, 400, 403			1244
.....	1716, 2151, 2154	\AMC@smashboxheight		\AMC@sza@callout	885, 886,
\AMC@numerotation		..	394, 397, 398, 399, 402		1257, 1262, 1266, 1268
.....	308, 312, 313	\AMC@smashcentered		\AMC@sza@callout@margin	1247
\AMC@outside@sep	..	..	395, 437, 438, 470, 471	\AMC@sza@callout@margin	
\AMC@oval@radius		\AMC@SR			1251
....	449, 523, 540, 542		258,	\AMC@sza@callout@none	1239
\AMC@ovalbox@	463, 477, 2684		260, 263, 264, 265,	\AMC@sza@callout@question	
\AMC@ovalbox@F	464, 475, 2685		266, 268, 270, 271, 2565		1243
\AMC@ovalbox@R		\AMC@SR@count		\AMC@sza@depth	
.....	460, 461, 483, 2682		264, 266, 274, 278,		1236, 1256, 1261
\AMC@ovalbox@RF	462, 481, 2683		281, 284, 285, 286, 287	\AMC@sza@height	
\AMC@pageHook	2492, 2499, 2500	\AMC@SR@time			1236, 1255, 1260
\AMC@pagepos			275, 2633	\AMC@sza@init@margin	1246
\AMC@premierecopie		\AMC@SR@advance	261, 268, 269	\AMC@sza@init@margin	1250
..	630, 2555, 2558, 2652	\AMC@SR@bit		\AMC@sza@init@none	1238
\AMC@prepare	..	\AMC@SR@const	..	\AMC@sza@init@question	1242
\AMC@prepare@element	...	\AMC@SR@max		\AMC@sza@width	
.....	705, 713, 828	\AMC@SR@nextByte			1236, 1254, 1259
\AMC@printformoutside@false		\AMC@SR@num	276, 277, 279,	\AMC@tempenv	
.....	553		280, 284, 287, 302, 303		686, 687
\AMC@printformoutside@true		\AMC@SR@set	260, 272, 273, 275	\AMC@tf@colW	
.....	555, 556	\AMC@SR@test		..	1069, 1070, 1073, 1079
\AMC@printkeyoutside@false		\AMC@SR@value		\AMC@tf@idtext	
.....	570	\AMC@stepQuestion		..	1050, 1055, 1072, 1075
\AMC@printkeyoutside@true			1282, 1317, 1357	\AMC@tf@idtitle	
.....	572, 574	\AMC@sti	289, 297, 301, 304, 305		1028, 1051, 1056
\AMC@qaff		\AMC@stil		\AMC@tf@nanswers	1053,
.....	880,		290,		1066, 1067, 1079, 1090
.....	1282, 1325, 1326, 1360	\AMC@surveyfalse		\AMC@tf@ncols	1046, 1054,
\AMC@randomseeds			2661		1078, 1086, 1090, 1094
..	2796, 2800, 2804, 2812	\AMC@surveytrue		\AMC@tfaw	
\AMC@randomseeds@	2788, 2797		1201,	..	1049, 1082, 1084, 1085
\AMC@save@box	390, 445, 2679		1203, 1205, 1208, 1251		

\AMC@tfbox . 1052, 1075, 1076	\AMCbin@zero . . 652, 671, 676	\AMCformAfterQuestion . .
\AMC@tfcolw . . 1048, 1079,	\AMCbloc 1298	 878, 1332
1080, 1081, 1084, 1085	\AMCbo@help . 2416, 2417, 2430	\AMCformAnswer 877, 1032, 1073
\AMC@tracebox	\AMCboHide 2443, 2449	\AMCformAnswerA . . . 894, 998
. 322, 376, 469, 495,	\AMCboOpts 2420	\AMCformatChoices
613, 615, 618, 1150, 2475	\AMCboShow 2421, 2449	. 1521, 2260, 2406, 2449
\AMC@tracechar 364, 566, 584	\AMCbotextGoto . . 2415, 2444	\AMCformBeforeQuestion .
\AMC@tracepos	\AMCboxColor 547, 2513	 877, 883, 1071
..... 323, 434, 440,	\AMCboxDimensions . 548, 665	\AMCformBegin . 16, 895, 2885
1136, 1139, 1323, 1332	\AMCBoxedAnswers 1109, 2624	\AMCformFilter 1016
\AMC@traceposx . . . 332, 352	\AMCBoxOnly 2415	\AMCformHSpace . . 69, 893,
\AMC@traceposy 341, 351, 353	\AMCboxOutsideLetter . .	1028, 1069, 1072, 2427
\AMC@unnumero 308	 549, 2359, 2428	\AMCformQuestion . 877, 1072
\AMC@use@box 391,	\AMCboxStyle	\AMCformQuestionA 881, 1005
475, 477, 481, 483, 2680	 21, 518, 2624, 2671	\AMCformQuestionN . 880, 888
\AMC@VERSION 2855, 2857	\AMCccircle 2471, 2474	\AMCformS 895
\AMC@watertext . . 2452, 2463	\AMCchoiceLabel	\AMCformTH
\AMC@XYFILE 325, 334, 343,	. . . 507, 588, 1032, 1095	. . 1028, 1047, 1057,
355, 360, 361, 362,	\AMCchoiceLabelFormat . .	1072, 1074, 1076, 1080
366, 372, 374, 1303,	 421, 507	\AMCformVSpace
1326, 1440, 2564,	\AMCcleardoublepage 16, 2591	 69, 877, 1071, 1091
2608, 2613, 2614,	\AMCcode 1517	\AMCgroup@pre 771, 794
2853, 2854, 2855,	\AMCcodeGrid 15, 1378	\AMCgrouploop@next
2856, 2857, 2858,	\AMCcodeGridInt . . 15, 1378	 796, 809, 827
2859, 2860, 2861,	\AMCcodeH 1518	\AMCgrouploop@prep
2862, 2863, 2865, 2867	\AMCcodeHspace	 782, 807, 825
\AMCaddpagesto 2582	 1379, 1420, 1479	\AMCgrouppre@cyclic . . 765
\AMCanswer	\AMCcodeID 1384, 1459	\AMCgrouppre@fixed . . . 727
. 1154, 1155, 2358, 2427	\AMCcodeID@@	\AMCgrouppre@withoutreplacement
\AMCassociation 2606	. 1383, 1387, 2660, 2823	 743
\AMCbeforeQuestion	\AMCcodeID@dot 1382	\AMCgrouppre@withreplacement
..... 1291, 1319, 1400	\AMCcodeID@mode . 1385, 2856	 734
\AMCbeginAnswer	\AMCcodeID@squarebrackets	\AMChorizAnswerSep
. 1126, 1155, 2357, 2426	 1381	 1144, 1145, 1153
\AMCbeginQuestion	\AMCcodeVspace	\AMChorizBoxSep
..... 1291, 1325,	. 1380, 1418, 1478, 1481	. 1146, 1147, 1150, 1152
1340, 1346, 1360, 1399	\AMCcompleteMulti . . 12, 66	\AMCid@check
\AMCbin@begin 680	\AMCcurrentenv 684, 687, 1316	29, 357, 2487, 2494,
\AMCbin@did 635, 643, 646,	\AMCdebutFormulaire 21, 2885	2495, 2514, 2518, 2847
648, 653, 656, 658, 664	\AMCdecimalPoint	\AMCid@checkmax
\AMCbin@digit	 1646, 1901, 1942	. . 625, 2495, 2650, 2847
. 633, 666, 669, 675, 678	\AMCdefault@groupmode . .	\AMCid@etud . 30, 319, 327,
\AMCbin@hsep 637, 640, 644, 654	 702, 716, 717	336, 345, 357, 360,
\AMCbin@id 634,	\AMCdontAnnotate . 360, 2374	361, 362, 367, 382,
639, 647, 648, 657, 658	\AMCdontScan 360, 2372	503, 1326, 2484, 2485,
\AMCbin@endigits	\AMCdump@reponses . 850, 864	2514, 2518, 2526,
..... 636, 638, 675, 678	\AMCemptybox 1176, 1201, 1236	2554, 2555, 2557,
\AMCbin@number	\AMCendAnswer	2565, 2570, 2577,
. 632, 665, 668, 670, 672	. 1127, 1155, 2357, 2426	2578, 2598, 2599, 2608
\AMCbin@one 642, 670	\AMCexponent 1648, 2219, 2231	\AMCid@etudfin 32,
\AMCbin@sequence . . 631,	\AMCexternalQuestion . . 1307	2556, 2557, 2558, 2577
665, 670, 671, 676, 677	\AMCfilterRefs 246	\AMCid@etudstart
\AMCbin@style 641, 665	\AMCform . 16, 895, 1096, 2884	 31, 2554, 2578

\AMCid@name	68,	1312, 1332, 1336,	\AMCnumericChoicesPlain
594, 597, 599, 603,		1341, 1347, 1374,	2258, 2265
605, 947, 1004, 1136,		1375, 1376, 1377,	\AMCnumericHide
1139, 1150, 1300,		1415, 1529, 1744,	2255, 2260
1303, 1332, 1416,		1777, 1780, 1786,	\AMCnumericOpts
1603, 1747, 1750, 2301		1789, 1795, 1798,	1740
\AMCid@quest	28,	1807, 1820, 1867,	\AMCnumericShow
360, 361, 362, 594,		1891, 2012, 2015,	1283, 2569
597, 599, 603, 605,		2048, 2050, 2070,	\AMCocol@Background
613, 615, 618, 946,		2140, 2147, 2164,	2291, 2298, 2363
1004, 1136, 1139,		2166, 2169, 2171,	\AMCocol@BoxFrameRule
1150, 1300, 1302,		2174, 2176, 2179,	2292, 2316, 2363
1303, 1323, 1326,		2183, 2186, 2188,	\AMCocol@Foreground
1332, 1416, 1748,		2195, 2200, 2202,	2294, 2299,
1751, 2070, 2340, 2423		2242, 2340, 2397,	2368, 2369, 2379, 2382
\AMCIDBoxesA	2485, 2489	2423, 2440, 2552,	\AMCocol@FrameRule
\AMCIDBoxesABC	2488, 2515	2570, 2670, 2674,	2293, 2317, 2390
\AMCIDBoxesB	2486, 2490	2721, 2790, 2793,	\AMCOpen
\AMCIDBoxesC	2487, 2490	2848, 2849, 2850, 2851	2289
\AMCids@height	80, 91, 2514	\AMCnarray@Border	\AMCopen@answer
\AMCids@sidefalse	83, 85	1652, 1846, 1847, 1991	2295, 2302, 2331, 2347
\AMCids@sidetrue	87	\AMCnarray@BorderWidth	\AMCopen@boxframerule
\AMCids@topfalse	83, 87	1654, 1844, 1989	2314, 2315, 2362
\AMCids@toptrue	85	\AMCncol@Background	\AMCopen@boxmargin
\AMCids@width	79, 90, 2517	1660, 1846, 1991, 2211	2312, 2313, 2361
\AMCidsPosition	77	\AMCncol@Border	\AMCopen@contentcommand
\AMCidsVar	81	\AMCncol@BorderWidth	2303, 2391
\AMCidsVarN	81, 82	1658, 2210	\AMCopen@framerule
\AMCif@env	685, 1364, 1366	\AMCneeds@storeboxfalse 2657	2318, 2319, 2389
\AMCif@category	979	\AMCneeds@storeboxtrue 2658	\AMCopen@Hspace
\AMCinterBquest	74, 1332	\AMCnobloc	2304,
\AMCinterBrep	69, 1140	\AMCnoCompleteMulti	2305, 2360, 2365, 2377
\AMCinterIquest	73, 1332	12, 67, 2356	\AMCopen@LineHeight
\AMCinterIrep	69, 1117	\AMCNombreCopies	2308, 2309, 2330
\AMCintervalFormat 2271, 2282		2550, 2551, 2864, 2867	\AMCopen@Lines
\AMCintervals 17, 2271, 2887		\AMCnoScoreZone	2310, 2311, 2335
\AMClabel	2599, 2604	1281, 1401	\AMCopen@lines
\AMCload@@reponse	861,	\AMCnontextGoto	2328
863, 1169, 1174, 2723		1637, 2251	\AMCopen@lineuptext 2297,
\AMCload@counter		\AMCnontextSign 1644, 1966, 1967	2320, 2345, 2348, 2351
27, 839, 840, 841,		\AMCnontextVHead	\AMCopen@question
842, 845, 846, 850,		1638, 1905, 2080	2296, 2301, 2364, 2365
853, 854, 855, 856, 859		\AMCnum@copies	\AMCopen@Width
\AMCload@reponse	844, 863	2551, 2552, 2553, 2556	2306, 2307, 2329
\AMClocalized	95	\AMCnum@questions	\AMCopenHide
\AMCloop@k	748, 754,	\AMCnumeric@Hspace	2400, 2406
758, 760, 781, 789, 798		1639, 1651, 1808,	\AMCopenOpts
\AMCmarginNote		1810, 1901, 1929,	2327
1182, 1208, 1251		1985, 1987, 1994, 2231	\AMCopenShow
\AMCmem@elt@cat 953, 965, 969		\AMCnumeric@Vspace	2338, 2406
\AMCmessage	19, 309,	1640, 1650,	\AMCOpenX
621, 1023, 1167, 1173,		1776, 1785, 1794,	2413
1302, 1310, 1311,		1822, 1893, 1907,	\AMCotextGoto
		1958, 1967, 2218, 2220	2289, 2401
		\AMCnumericChoices	\AMCotextReserved 2290,
		18, 1521	2300, 2376, 2380, 2382
			\AMCoutsideLabelFormat 549
			\AMCPackageOptionsExterne
			2663, 2664
			\AMCpageref
			2602
			\AMCpostNquest
			75, 2240
			\AMCpostOquest
			76, 2393
			\AMCqlabel
			2603

913, 915, 918, 921,
924, 928, 933, 935,
938, 940, 943, 945,
949, 950, 952, 955,
956, 958, 961, 962,
964, 967, 976, 977,
978, 979, 1398, 1408,
1413, 1438, 1443,
1445, 1448, 1450,
1490, 1494, 1499,
1504, 1509, 1512,
1515, 1516, 1517,
1518, 1536, 1544,
1554, 1557, 1562,
1566, 1578, 1581,
1583, 1586, 1588,
1594, 1607, 1612,
1624, 1728, 1732,
1740, 1742, 1756,
1771, 1802, 1813,
1834, 1837, 1839,
1883, 1897, 1936,
1964, 1980, 2006,
2068, 2247, 2249,
2255, 2264, 2267,
2268, 2276, 2286,
2409, 2412, 2413,
2770, 2771, 2773,
2774, 2776, 2777,
2780, 2781, 2783, 2784

\csvreader
 . 2695, 2700, 2706, 2715

D

\define@boolkey 533, 1060,
 1061, 2321, 2322,
 2323, 2324, 2325, 2418

\define@choicekey
 81, 525, 1219, 1264

digits (option) 10

\ding 497

\dontannotate 361

\dontscan 360

\dotfill 385, 2333, 2349

\draw . . . 449, 453, 1195, 1197

E

\element 14, 693, 2894

\endchoices 2870

\endchoicescustom 2872

\endchoiceshoriz 2871

\endcopieexamen . . 2880, 2893

\endexamcopy 2880, 2892

\endreponses 2870

\endreponseshoriz 2871

\endresponsesperso 2872

ensemble (option) 21

environments:

amcxyfile 371

auto 2688

choices 12, 1109

choicescustom . . 12, 1109

choiceshoriz . . . 12, 1109

copieexamen 21

examcopy 2760

question 11, 1296

question-auto 2688

questionmult . . . 11, 1296

Questionnaires 2688

questionouverte . . . 1296

reponses 21

reponseshoriz 21

reponsesperso 21

tikz-multi 1109

tikz-single 1109

values 2688

values-auto 2688

variable-auto 2688

variable-multi 1296

variable-single . . . 1296

\evensidemargin . 1189, 1191

\examcopy 2880, 2890

examcopy (env.) 2760

\exemplaire 21, 2878

\exemplairepair 2598

\exp 250

\explain 1362, 2750

\explaincontext 1369

\ExplSyntaxOff 257,
 981, 1520, 1636, 2257,
 2270, 2288, 2414, 2787

\ExplSyntaxOn 232,
 895, 1388, 1534, 1641,
 2262, 2272, 2408, 2768

F

\fancypagestyle
 . 2501, 2506, 2528, 2535

\fbox 2526

\fmtversion 3, 2498

\footrulewidth
 . 2504, 2523, 2533, 2538

\foreach 2714

\formulaire 21, 2884

\fp 1548, 1556, 1558, 1560,
 1564, 1569, 1571,
 1573, 1577, 1584,
 1589, 1596, 1613,

1616, 2057, 2058,
2096, 2097, 2106,
2114, 2116, 2139,
2140, 2166, 2274,
2275, 2277, 2278,
2279, 2280, 2282, 2283

français (option) 10

fullgroups (option) 11

G

\group 1540, 1552

H

\he@dbas . . . 2479, 2508, 2509

\he@dhaut 2480, 2512

\he@dtaille . 2476, 2479, 2480

\hfuzz 399

\ht 397, 1177

I

\ifAMC@affichekeys
 38, 1324, 1325

\ifAMC@asqbloc . 44, 878, 884

\ifAMC@automarks . . . 63,
 1010, 2568, 2587, 2594

\ifAMC@calibration
 48, 324, 333,
 342, 355, 360, 361,
 362, 365, 1167, 1173,
 1301, 1309, 1326,
 1335, 1374, 1375,
 1376, 1377, 1415,
 1439, 2564, 2570,
 2607, 2612, 2721, 2852

\ifAMC@catalog 49, 416, 2825

\ifAMC@correc 40,
 408, 1169, 1363, 1370,
 1746, 2331, 2347, 2723

\ifAMC@correthead
 37, 2496,
 2542, 2558, 2833, 2839

\ifAMC@draw@cross
 392, 431, 452

\ifAMC@ensemble 57,
 554, 571, 592, 611,
 898, 1010, 1013,
 1017, 1022, 1325,
 1527, 2069, 2240,
 2241, 2339, 2344,
 2358, 2364, 2396,
 2422, 2427, 2430,
 2439, 2568, 2848, 2858

\ifAMC@extractOnly
 42, 2208, 2862

noshuffle (option)	10	\pdfpagewidth	359, 2780, 2781	250, 1391, 1465, 1473,	
noshufflegroups (option)	10	\pgfextractx	1185	1539, 1547, 1548, 1549	
\nouveau groupe	693, 818	\pgfm@rnd@z	2789, 2792	\setdefaultgroupmode	716
nowatermark (option)	4, 10	\pgfm@parse	2792	\setgroupmode	702, 716
\NoWatermarkExterne		\pgfm@result	2793	\shuffle@it	850, 864
.	2633, 2788	\pgfp@ntanchor	1186	\shufflegroup	
		\pgfp@ntdiff	1185		14, 740, 752, 774, 2881
O		\pgfp@ntorigin	1185	\shufflegroupslice	
\oddsidemargin	1188, 1192	plain (option)	11		757, 760, 777
\onecopy	2549,	postcorrect (option)	11	\smash	2331, 2347
.	2690, 2764, 2878, 2889	\prg	897, 900,	\space	2750
options:			902, 905, 1501, 1597, 1609	\start@Answers	1110, 1305
answers	10			\storebox	2679
asbox	10	Q		storebox (option)	11
automarks	11	\question	1296, 1303	\str	965
bloc	21	question (env.)	11, 1296	\strut	1195, 1197
box	10	question-auto (env.)	2688	\SujetExterne	2788
calibration	5, 10	\QuestionIndicative	13,	\SurveySingleAnswerfalse	
catalog	10		1341, 1347, 1374, 1409		1338
chiffres	21	questionmult (env.)	11, 1296	T	
completemulti	10	Questionnaires (env.)	2688	\tabcolsep	1847
correc	21	\questionnum	1326	\tex	1542
correcindiv	21	questionouverte (env.)	1296	\textasciicircum	1649
digits	10	R		\TextField	382
ensemble	21	\raggedcolumns	1093	\textit	103,
francais	10	\raisebox	1647		116, 143, 157, 196, 222
fullgroups	11	\refstepcounter	1289	\textsc	2502
indivanswers	6, 10	\reponses	2870	\textsf	534
init	10	reponses (env.)	21	\the@cs@vrow	
insidebox	10	\reponseshoriz	2871		1141, 1142, 2729, 2738
keys	10	reponseshoriz (env.)	21	tikz-multi (env.)	1109
nopage	6, 11	\reponsesperso	2872	tikz-single (env.)	1109
noshuffle	10	reponsesperso (env.)	21	\tikzset	2727
noshufflegroups	10	\restituegroupe	21, 2882	\times	1649
nowatermark	4, 10	\retick	362	\tl	930, 931, 936,
ordre	21	\rngstate	2565		941, 970, 972, 1536,
outsidebox	10	S			1538, 1545, 1546,
pdfform	11	\savebox	390		1755, 1758, 1764,
plain	11	\savepos	2771		1767, 1832, 1840,
postcorrect	11	\sbox	1177		1842, 1848, 2008,
separateanswersheet		\scoring	13,		2022, 2024, 2027,
.	5, 10, 16		1341, 1347, 1374, 2875		2028, 2032, 2037,
storebox	11	\scoringDefaultM			2038, 2040, 2043,
\or	84, 86		13, 1374, 2876		2044, 2052, 2064, 2065
ordre (option)	21	\scoringDefaultS		\tracepos	326, 335, 344
outsidebox (option)	10		13, 1374, 2877	U	
\ouverte@vs	1296	\ScoringExterne	2788	\une@rep	1115, 1116,
P		\section	1100, 1104		1121, 1125, 1129,
\PackageError		separateanswersheet (op-			1132, 1169, 1174, 2723
.	678, 720, 791, 2749	tion)	5, 10, 16	\useasboundingbox	447
\pageref	2602	\seq	234, 235,	\usebox	391
pdfform (option)	11		238, 239, 240, 243,	\usestorebox	2680
\pdfpageheight	359, 2783, 2784				

<code>\usetikzlibrary</code>	2726	<code>\version</code>	2855	2859, 2860, 2861,
		<code>\vfuzz</code>	399	2862, 2863, 2865, 2867
V		<code>\vrule</code>	1986	<code>\wrongchoice</code>
<code>\value</code>	1033	<code>\vtop</code>	1468	13, 874, 875, <u>1158</u> , 2281, 2874
<code>values (env.)</code>	<u>2688</u>	W		
<code>values-auto (env.)</code>	<u>2688</u>	X		
<code>variable-auto (env.)</code> . . .	<u>2688</u>	<code>\wd</code>	399, 400, 1076, 1177	<code>\x</code>
<code>variable-multi (env.)</code> . .	<u>1296</u>	<code>\with</code> . . .	1440, 2613, 2614,	<code>\xyclose</code>
<code>variable-single (env.)</code> .	<u>1296</u>		2856, 2857, 2858,	<code>\xyopen</code>
				374
				372