Integrating LaTeX and Moodle Questionnaires

L. Garcia-Forte, C. Leon-Hernandez and C. Rodriguez-Leon

Email	lgforte@ull.es, cleon@ull.es, casiano@ull.es
Website	http://nereida.deioc.ull.es
Address	Dpto. Estadística, I.O. y Computación
	Universidad de La Laguna
	Tenerife, Spain

Introduction
Preparing and...
Automatic Solution
Conclusions
Acknowledgments

Home Page

Title Page

◀◀   ▶▶

◀   ▶

Page 1 of 27

Go Back

Full Screen

Close

Quit

Introduction

Preparing and . . .

Automatic Solution

Conclusions

Acknowledgments

Home Page

Title Page

◀◀   ▶▶

◀   ▶

Page 2 of 27

Go Back

Full Screen

Close

Quit

Abstract   The manufacturing of teaching material conveys the generation of both static (unreactive) data-documents and dynamic (reactive) program-documents based on different technologies. Teaching a subject often implies the maintenance of a large number of both types of documents, usually written in a variety of languages and stored in diferent formats. Ergo a natural goal for the lecturer is to minimize the amount of work invested during the development and maintenance of the material. There are acceptable solutions regarding the transformation between formats with the same kind of reactivity. This work discusses the problem of integrating Moodle (a Open Source Learning Management System) and LaTeX (a document preparation system), proposes a methodology to pursuit this goal and presents a tool to assist in the translation of Moodle Quiz documents to LaTeX.

1.   Introduction

Like the majority of university people working inside the scientific/mathematic scope, our usual environment for the development of documents is LaTeX. This is a formidable mark-up language based on TeX. TeX was designed by Donald Knuth and is firmly settled among the scientific community. The principal difference between LaTeX and other programs like Word is that LaTeX is a *document processor* rather than a *docu-*

ment editor. A LaTeX document must be compiled with a LaTeX compiler to produce the target format. The LaTeX family of tools is remarkable efficient for the preparation of scientific and technical documents. Consequently, it is by using these tools that we generate the non reactive documents for the students. A large number of editors have developed their own LaTeX styles for the publication of journals and books. With LaTeX any kind of document like books, articles, reports and slides combining text, equations, tables, figures, graphics, bibliography, etc. can be prepared. From a LaTeX document, it is straightforward to generate files in various formats, for example, DVI, Postscript, PDF, HTML, etc., using the proper translators LaTeX, `dvips`, `pdflatex`, `latex2html`, etc. respectively [1].

However, the trend during the recent years is to add to the traditional and efficient chalk-blackboard approach the use of software tools for the publication of notes, exercises, exams, slides, transparencies, etc. The general direction is to move towards software environments and tools giving support, promoting and easing a bidirectional communication among all the participants, in which students have more opportunities to be active. From this perspective, we can differentiate between reactive and unreactive documents. The manufacturing of teaching materials conveys the generation of passive, unreactive data-documents, in passive unreactive formats: LaTeX, Postscript, .doc, PDF,

HTML, XML, gift, png, eps, etc. It also implies the production of active, reactive program-documents based on different technologies: Java, PHP, CGI, HTML+JavaScript, Perl, Pyton, MySQL, PostGres, etc. These tools - being generic - are not appropriate for their direct use inside the education process. New teaching high-level languages and tools are required. Tools like Moodle [2, 3, 4] or ATutor [5] provide forums, chats, user management (differentiating between students and lecturers), work groups, workshops, interactive exercises, polls, calendars, tasks, etc. The interaction is not restricted to the student-lecturer relation but it also facilitates the opportunities for interaction among the students. Moodle is a course management system. Its name stands for Modular Object Oriented Dynamic Learning Environment. It is a free, Open Source software package designed to help educators create effective online learning communities. The underlying philosophy behind Moodle is that learning is a process strongly bounded to our experiences and that learning occurs particularly well when working in a collaborative environment.

The creation of both reactive and unreactive documents associated with the teaching of some subject includes the maintenance of a large number of files written in a variety of languages and perhaps stored in diferent formats. It is obvious that we, lecturers, want to minimize the amount of work invested during the development and maintenance of

the material. The usual strategy that we all follow is to keep a reduced set of source documents, preferably written in a reduced number of languages, and to rely on software tools to generate all the target formats. There are acceptable solutions regarding the transformation between unreactive formats: It is enough proof to remember the innumerable family of xxx2yyy format translators of which LATEX2HTML is a good example. Also, most Open Source Learning Management Systems - like Moodle - give satisfactory support for translations between reactive formats.

This work discusses the problem of integrating Moodle Quiz module and LATEX, examines the existent solutions, proposes a methodology to pursuit this goal and presents a tool to assist in the translation of exercise sheets as a small step towards lecturers chimerical search for document singleness.

The contents of this contribution are organized as follows: The next section presents the methodology we currently use to integrate LATEX and Moodle documents. The third section describes the use of a translator from GIFT - one of the Moodle formats for the representation of exercises - to LATEX which substantiates some facets of our proposal. The last section attempts to summarize our conclusions and foresees our future works in this direction.
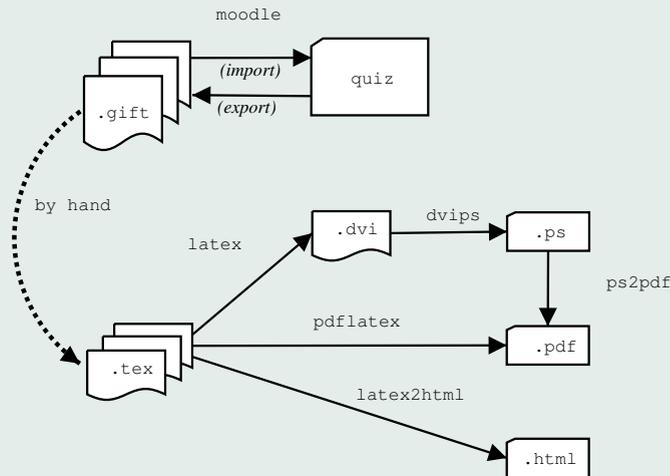
Figure 1: Scheme of the methodology for the document generation

2.   Preparing and Maintaining Questionnaires

Moodle has been adopted by our "Escuela Técnica Superior de Ingeniería Informática de la Universidad de La Laguna", the place where we teach [6] subjects like Compilers, Programming, OOP, Parallel Programming, etc. Therefore, we had to migrate and integrate the existing

Home Page

Title Page

◀◀   ▶▶

◀   ▶

Page 6 of 27

Go Back

Full Screen

Close

Quit

material to the new platform. The part requiring major effort was the preparation of questionnaires and problem sheets (See Figure 1). It was necessary to export, translate and import the interactive questionnaires to have both reactive and unreactive versions of them.

A warning about the use of electronic questionnaires. When dealing with the evaluation of our students we usually differentiate between two kinds of evaluation:

– Additive Evaluation: The information is used to grade and credit the student certificating the level of competentece reached.

– Formative Evaluation: the information is used to guide and improve the learning process

The most important attribute of the Formative Evaluation is its capacity to provide timely information even if not as accurately as we can expect for Additive Evaluation. It is for this purpose that the use of electronic questionnaires and multiple-choice type exams is valuable.

To better illustrate the methodology for the preparation of documents, let us consider the make up of an exercise sheet for autoevaluation.

Home Page

Title Page

◀◀ ▶▶

◀ ▶

Page 7 of 27

Go Back

Full Screen

Close

Quit

Step 1. Preparing the Quiz

The easiest way to write a questionnaire is to take advantage of the Moodle interface for building questionnaires. That is straightforward to use, but, depending on the circumstances, it can be unbearably slow. After filling the corresponding forms you can export the questionnaire to GIFT format.

According to the Moodle manual, GIFT is the most comprehensive import / export format available for importing quiz questions from/to a text file. It supports Multiple-Choice, True-False, Short Answer, Matching and Numerical questions. Various question-types can be mixed in a single text file, and the format also supports line comments, question names, feedback for the student and percentage-weight grades.

To export the questionnaire, select "Questionnaire". Once in the edition window, choose the option "Export questions to file" . That will open a new window. Choose the format and the name of the file. By default the file is stored in the server inside the course subdirectory `questionnaire`.

Figure 2 shows the result. We'll keep working with this example along this article. Observe that questions are delimited by a double carriage-return. The first question (lines 1-7) is an example of a TRUE-FALSE question. Each question is divided into three sections: The state-

ment prefix, the answer section bounded by the pair of non escaped curly brackets and the statement postfix. Comments, like the one in line 1, start with two slashes and expand to the end of the line. The prefix occupies lines from 2 to 7. The text between pairs of colons is the question identifier. Next, between brackets comes the format used for this question. These two fields are optional. The source format in the example is HTML expanded with TeX. This is achieved through the TeX filter MimeTeX. The MimeTeX filter extends Moodle with the capacity of managing TeX formulas ([7, 8]). MimeTeX is always in math-like mode. Any text between \$\$ delimiters is interpreted by Moodle as a math formula and sent to the filter which renders the corresponding image. It may optionally start with the name of the question enclosed by a pair of colons and the format (HTML, PLAIN, etc.). When the TeX filter is installed the text is a combination of HTML and LaTeX. The correct answer appears between braces in line 7. LaTeX formulas can also be used in this section. Lines 9-21 present a MULTIPLE-OPTION question. The correct answer is prefixed by the symbol "=" while wrong answers have a "~" at the beginning.

An alternative and faster way to the Moodle interface is to directly write the GIFT with your document favourite editor. This approach makes easier the recycling of existing LaTeX and HTML material and may be automatized through somewhat ad-hoc scripts. It also allows

for a more "relaxed" and human readable GIFT files.

Figure 3 shows a typical "human edited" version of the example in Figure 2. The file appears less cluttered among other reasons because GIFT metasymbols (like =, ~, {,}, etc.) aren't escaped. The escape is done through a utility script acompanying the software presented in Section 3.

If we choose the last option, we have to import/upload the GIFT file to the Moodle site. This also gives a path to migrate the exercises to another courses or another Moodle installation. To import a quiz, edit an activity of type "questionnaire" and once in the edition window (Figure 4) select the option "Import questions from file". A window like the one in Figure 5 is open. From there we can proceed to locate the file and to upload it. The result of a succesful upload is displayed in Figure 6.

Step 2. Preparing the Material in LaTeX

To obtain equivalent high-quality non-reactive formats we translate the GIFT file into a new LaTeX file. Figure 7 shows a human-made direct translation to LaTeX where two nested enumerate environments have been used. From the LaTeX version we can easily obtain postcript using `dvips`, PDF using `pspdf` or `pdflatex` and HTML using `latex2html`.

Though they were actually obtained using the tool to be described next in Section 3, Figures 8 and 9 give you an idea of the final appearance.

3.   Automatic Solution

Starting from an interactive quiz and following the steps described in the former section there is always a mean to produce a file in GIFT format describing the questions. The translation is made by a Perl [9] program named gift2latex. Figure 10 outlines the process. The first step to obtain one of the unreactive formats from the quiz (PDF, Postscript, etc.) is to export the quiz to GIFT format. As is described in the previous section, this transformation is bidirectional, that is, Moodle allows both import and export operations on GIFT files. From this source, the script gift2latex produces two LATEX files; Each one contains a LATEX section. The first one describes the questions. The second (referenced by the former) describes the answers. However, depending on the execution options the output can be a standalone full latex document or the two files describing the questions and answers, to be included inside a main document.

Let us assume the quiz shown in Figure 2 is stored in a file named exercises. gift. Figures 11 and 12 show fragments of the ouputs

obtained when executing the command line:

```
$ gift2latex exercises.gift
```

the two generated files (named exercises_questions.tex and exercises_answers.tex) can then be embedded inside a main document using the LATEX \input command.

Observe that the LATEX code in Figure 11 is more sophisticated than the one in Figure 7. Links between each pair of question-answer items are generated.

Figures 8 and 9 showed a visual sample of the result of compiling with LATEX and LATEX2HTML the LATEX files generated by GIFT2LATEX. The HTML version is navigable: Clicking the question buttons (left figure) takes you to the corresponding item inside the Answer section (right).

Teachers can change the style of the output modifying the coresponding style files, usually found in the etc distribution directory. There are a couple of style files per type of question. They control the output aspects for the question and answer sections. The syntax to describe a style is a mixture of LATEX and Perl. Figure 13 presents a fragment of a translation template or style file for TRUE-FALSE questions. Chunks of text between the %< and %> correspond to the variable part (Perl code) inside the fix LATEX structure.

Home Page

Title Page

◀◀ ▶▶

◀ ▶

Go Back

Full Screen

Close

Quit

4. Conclusions

This work discusses the problem of integrating Moodle and LaTeX documents proposes a methodology to solve it and presents a tool to assist in the translation of questionnaires. The steps to produce the materials are:

1. Write a questionnaire taking advantage of the Moodle interface for building questionnaires.

2. After filling the corresponding forms, export the questionnaire to GIFT format.

3. Run the script `gift2latex` on this file: it produces two LaTeX files; One describes the questions, the other the answers.

4. Finally, include these files in the main latex document.

Alternatively, since GIFT is more human friendly than other formats as XML, the lecturer can omit steps 1 and 2 and directly edit the questionnaire in GIFT formatusing her favourite editor.

The tool is currently a functional prototype and we expect it will be soon delivered in CPAN [10]. A version of the front-end GIFT parser is already there [11].

5. Acknowledgments

References

[1] N. Dragos. The latex2html translator. *http://www-exdev.mpce.mq.edu.au/l2h/docs/manual/*, 1999.

[2] Moodle manual. *http://www.moodle.org*.

[3] Moodle in the Wikipedia. *http://en.wikipedia.org/wiki/Moodle*.

[4] Joyce Seitzinger. Moodle Tool Guide for Teachers. *http://www.cats-pyjamas.net/wp-content/uploads/2010/05/MoodleToolGuideforTeachers_May2010_JS.pdf*.

[5] Atutor manual. *http://www.atutor.ca*.

[6] Escuela Técnica Superior de Ingeniería Informática. *http://www.etsii.ull.es*.

[7] John Forkosh. mimetex quickstart. *http://www.forkosh.com/mimetex.html*.

[8] Andreas Leiser. Math using TEX. *http://nereida.deioc.ull.es/html/moodle_help_tex.htm*.

[9] L. Wall, T. Christiansen, and R. Schwartz. *Programming Perl*. O'Reilly, 2nd edition, 1991.

[10] The Perl Community. CPAN: Comprehensive Perl Archive Network. *http://www.cpan.org/*.

[11] Casiano Rodriguez-Leon and Coromoto León and Luis García. The Gift Parser Module. *http://search.cpan.org/~{}casiano/Gift-0.6/*.

```
 1 // question: 401  name: FIRST
 2 ::FIRST::[html]Given a <i>grammar</i>
 3 $$G\=(\Sigma,V,P,S)$$ and  a
 4 <i>production</i> $$A \rightarrow \alpha$$
 5 it holds that
 6 $$FIRST(\alpha) \= \emptyset$$ implies
 7 $$A$$ is annullable?{FALSE}
 8
 9 // question: 402  name: accesing
10 ::accesing::[html]A multidimensional array in C
11 is simulated defining 1 dimensional arrays
12 whose elements are arrays. To compute the
13 relative position of one element
14 $$a[i_1, i_2, ..., i_k]$$
15 the following formula is applied:{
16  =$$(i_k + D_k(... (i_2 + i_1*D_2...))*size+
17   base-(L_k+D_k(... L_2+L_1*D_2...))*size$$
18  ~$$(i_k + D_k(...(i_3 + (i_2 + i_1*D_2)*D3)
19                ...)) * size + base$$
20  ~None of them
21 }
```

Figure 2: GIFT file generated by Moodle from a Quiz

```
Given a grammar
$$G =(\Sigma,V,P,S)$$
and  a production
$$A \rightarrow \alpha$$
does $$FIRST(\alpha) = \emptyset$$
implies $$A$$ is annullable?{FALSE}

A multidimensional array in C
is simulated defining 1-dimensional arrays
whose elements are arrays. To compute the
relative position of one element
    $$a[i_1, i_2, ..., i_k]$$
the following formula is applied:
{
 =$$(i_k + D_k(... (i_2 + i_1*D_2...))*size+
  base-(L_k+D_k(... L_2+L_1*D_2...))*size$$
 ~$$(i_k + D_k(...(i_3 + (i_2 + i_1*D_2)*D3)
              ...)) * size + base$$
 ~None of them
}
```

Figure 3: GIFT file. "Human Informal Style"
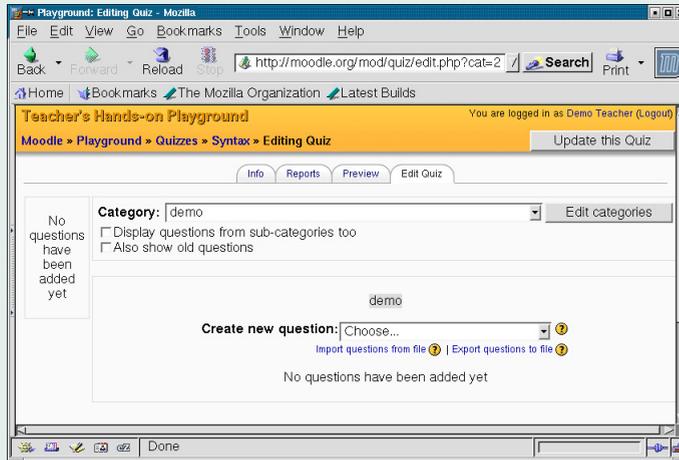
Figure 4: Editing a Quiz
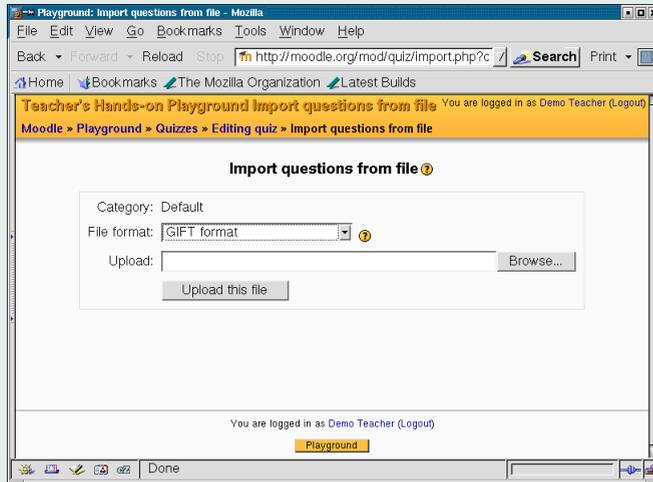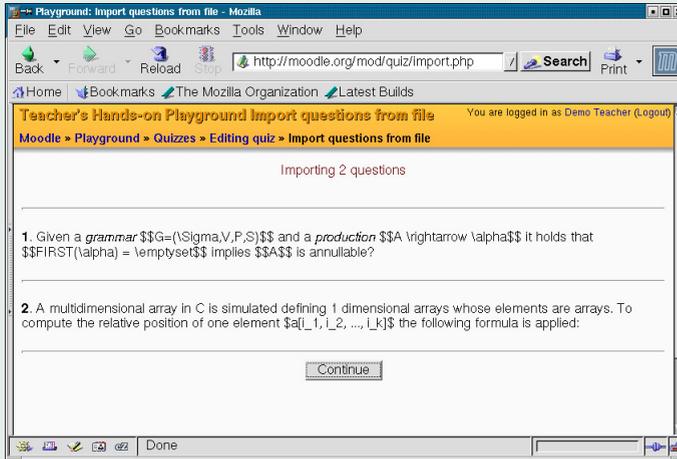
Figure 5: Loading the file

Figure 6: A GIFT file has been uploaded

```
 1 \begin{enumerate}
 2 \item
 3   Given a \emph{grammar} $G=(\Sigma,V,P,S)$ and
 4   a \emph{production} $A \rightarrow \alpha$ it
 5   holds that $FIRST(\alpha) = \emptyset$
 6   implies \emph{A} is annullable?
 7   \begin{enumerate}
 8     \item
 9       True
10     \item
11       False
12   \end{enumerate}
13
14 \item
15   A multidimensional array in C is simulated
16   defining 1 dimensional arrays whose elements
17   are arrays. To compute the relative position
18   of one element $a[i_1, i_2, ..., i_k]$ the
19   following formula is applied:
20   \begin{enumerate}
21     \item
22       $(i_k + D_k(... (i_2 + i_1*D_2...))*size+
23       base-(L_k+D_k(... L_2+L_1*D_2...))*size$
24     \item
25       $(i_k + D_k(...(i_3 + (i_2 +
26             i_1*D_2)*D3)...)) * size + base$
27     \item
28     None of them
29   \end{enumerate}
30 \end{enumerate}
```
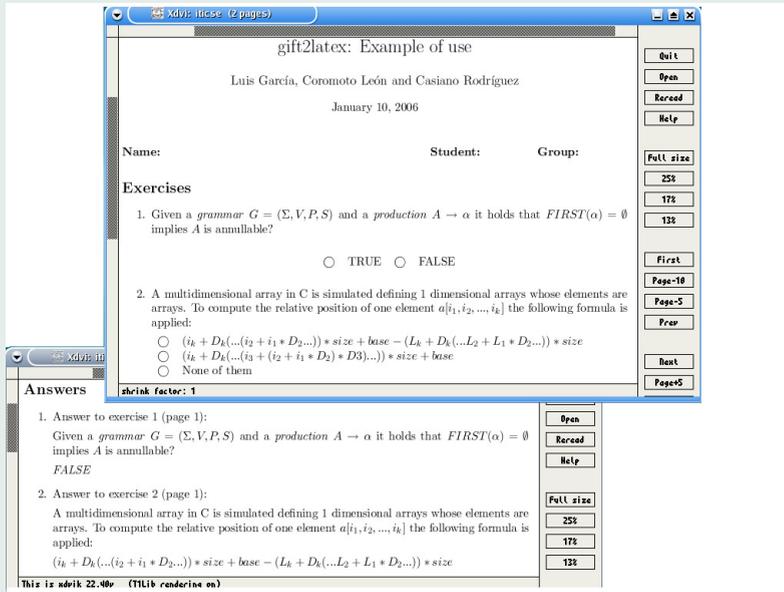
Home Page

Title Page

◀◀ | ▶▶

◀ | ▶

Page 21 of 27

Go Back

Full Screen

Close

Quit

Figure 8: Postscript generated from LaTeX

Home Page

Title Page

◀◀   ▶▶

◀   ▶

Page 22 of 27

Go Back

Full Screen

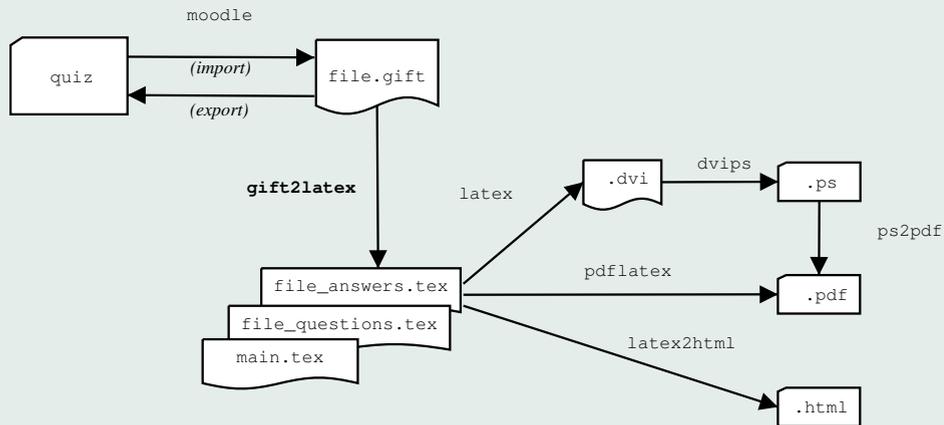Close

Quit

Figure 9: HTML generated by LATEX2HTML

Figure 10: Scheme using automatic translation

```
 1 \item
 2 \label{question:syntax1}
 3 Given a \emph{grammar}
 4  $G=(\Sigma,V,P,S)$ and a \emph{production}
 5  $A \rightarrow \alpha$ it holds that
 6  $FIRST(\alpha) = \emptyset$
 7  implies $A$ is annullable?
 8
 9 \begin{center}
10 \begin{tabular}{llll}
11 $\bigcirc$ & TRUE & $\bigcirc$ & FALSE
12 \end{tabular}
13
14 \noindent
15 \end{center}
```

Figure 11: Excerpt of the LATEX file for the question section

Introduction

Preparing and . . .

Automatic Solution

Conclusions

Acknowledgments

Home Page

Title Page

◀◀ ▶▶

◀ ▶

Page 26 of 27

Go Back

Full Screen

Close

Quit

```
 1 \item Answer to exercise
 2 \label{answer:syntax1}
 3 \ref{question:syntax1}
 4 (page
 5 \pageref{question:syntax1}):
 6
 7 \noindent Given a \emph{grammar}
 8  $G=(\Sigma,V,P,S)$ and a \emph{production}
 9  $A \rightarrow \alpha$ it holds that
10  $FIRST(\alpha) = \emptyset$
11  implies $A$ is annullable?
12
13 \emph{FALSE}
```

Figure 12: Excerpt of the LATEX file for the answer section

```
 1 %<$separator%>
 2 \label{question:%<$label%>}
 3 %<$prefix%>
 4
 5 \begin{center}
 6 \begin{tabular}{llll}
 7 $\bigcirc$ & TRUE & $\bigcirc$ & FALSE
 8 \end{tabular}
 9
10 \noindent %<$sufix%>
11 \end{center}
```

Figure 13: Excerpt of the template for TRUE-FALSE