The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
Paper ID #41930

# Improving Efficiency and Consistency of Student Learning Assessments: A New Framework Using LaTeX

**Dr. Ira Harkness, University of Florida**

Ira Harkness is an Instructional Assistant Professor in the Department of Materials Science and Engineering. He has two decades experience in higher education, including directing information technology and facilities efforts at UF, and working with non-profits and community organizations to address K-12 education. His expertise is in computational nuclear engineering and nuclear engineering education.

**Prof. Justin Watson**

©American Society for Engineering Education, 2024

**Improving Efficiency and Consistency of Student Learning Assessments:**
**A New Framework Using LaTeX**

## Abstract

Instructors often rely on What-You-See-Is-What-You-Get (WYSIWYG) word processor applications like Microsoft Word to create student learning assessments such as homework assignments and exams. The shortcomings in this traditional method led to the authors' motivation to develop a better method. This project focuses on the development, usage, evaluation, and dissemination of a student learning assessment system based on LaTeX, a document preparation system commonly used for scientific documents. We call it the Student Learning Assessment Modular System (SLAMS). With this system, educators can streamline the assessment creation process. They can standardize outputs using a library of individual problems with integrated solutions. The system outputs a high-quality PDF file with optimum hyphenations, line breaks, and page breaks. Equations, tables, and figures are consistently formatted. The system is modular and flexible to allow the user to use a subset of the features or add new features based on their individual needs.

## Introduction

We describe our experience creating and using our Student Learning Assessment Modular System (SLAMS) for various science and engineering courses. The inefficiency of managing diverse problems and solutions, coupled with the tedious task of copying and pasting between documents, highlighted the need for a better solution. Our aim was to provide high quality assignments to students while minimizing time spent on logistics.

Given the limitations of existing tools such as Microsoft Word, we chose to instead develop a new system for the following reasons:

- We wanted to manage problems and associated solutions in a library.
- We wanted the flexibility to use any of the features of LaTeX.
- We wanted to focus time on pedagogical innovation and direct student engagement rather than the details of formatting assessments.

Designing assessments requires the instructor to consider the purposes, format, and other details of the assignment [1]. Instructors often choose Microsoft Word for creating student learning assessments. Since it is a What-You-See-Is-What-You-Get (WYSIWYG) editor it allows them to see the final layout as they create the document, making it more user-friendly. Word offers an extensive range of formatting tools and templates. However, for documents requiring complex mathematical formulas or highly controlled typography, using Word can be time intensive.

Other work has investigated using technology for developing homework assignments. Santoro and Bilisoly found making algebra and statistics problems by hand to be time-consuming and error prone [2]. To solve this problem, they created Mathematica programs that generate HTML files that are ultimately imported into Blackboard Learn. Useful capabilities of this system include the ability to generate mathematically equivalent problems with multiple choice or fill-in-the-blank solutions. Whiting and Scott created *YStatTest* to generate unique problems and solutions using a MySQL database and the R programming language [3]. Their system included the ability to select problems based on keywords, categories, and historical difficulties. These solutions require special skill and knowledge and are application specific solutions. They do not generalize well to a broader class of assessments, and it is not clear if these are currently maintained.

The SLAMS framework is designed to be generalized, flexible and extendable to a broad range of assessment types. The remaining sections of this paper provide more details on SLAMS, including background on TeX and LaTeX, current features of SLAMS, the process of creating a single student learning assessment, and feedback from students and instructors.

**TeX and LaTeX**

TeX, written by Donald E. Knuth, is a typesetting system designed for producing complex mathematical and scientific documents with a high typographic quality. Unlike common word processors, TeX is not a WYSIWYG text editor. Instead, it utilizes a markup language where users write in plain text and add formatting commands, which are then typeset into a PDF document. This approach allows for precise control over layout and typography but does have a learning curve and is less intuitive than Microsoft Word or similar.

LaTeX, originally written by Leslie Lamport, is an advancement of the original TeX that provides a higher-level, descriptive markup language for document preparation. It introduces pre-defined templates and commands that simplify the creation of structured documents. LaTeX streamlines complex formatting tasks, enabling users to focus on content rather than layout intricacies.

LaTeX has been used for creating student learning assessments since the late 1990s and possibly earlier. Laura Taalman and Michael Kozdron created a website called "A Gentle Introduction to TeX" when they were in the Duke University Math Department circa 1999 [4]. The site provided Laura's LaTeX templates for quizzes and exams. Other TeX resources are also available, including Blaik Matthews file (testpoints.tex) that allows tracking points for multipart problems. Our framework builds upon this previous work by adding additional packages and an organizational structure.

**SLAMS Overview**

The motivation behind the development of SLAMS was the desire to have an assessment framework that made maintaining years of assessments easier. Instructors often rely on a patchwork method, organizing assignments and exams by semester, with individual files for each assignment. Instructors may have additional files that contain the solution, or a handwritten copy of the solution in a drawer. Trying to retrieve questions and answer combination from previous years can be difficult and time consuming. SLAMS organizes all assessments, figures, and questions under a single directory structure, as shown in Figure 1. The top-level directory contains each individual assessment type (exam, homework, quiz, etc.), figures, and problems. This organization allows any assessment document to include figures or problems from their respective directories, as shown in Figure 1. This structure is user configurable and flexible to accommodate different types of assessments and different organizational preferences.

The following sections describe the features of the individual files located in each of the directories.

**Features**

Current features of SLAMS include an automated method for calculating assessment point values, problem statement and solution in one file, the use of style templates for automatic formatting, and a modular flexible framework. The dissemination and maintenance of the SLAMS framework will be through the use of GitHub. This will also make it easy for others to contribute and extend the framework beyond its current capabilities.

To generate automatic test points, SLAMS makes use of the testpoints.tex file created by Matthews [4]. This allows the user to assign point values to individual problems and subparts of a problem. During typesetting, the subparts totals will be compared to the problem and the total point value for the assessment will be calculated. The system provides a warning in an output log if the point values are internally inconsistent, such as when a problem's total points don't match the subparts. This provides the user with an easy way to ensure that the appropriate points have been assigned. Testpoints is implanted into the framework using the \input command, as demonstrated in the code excerpt below.

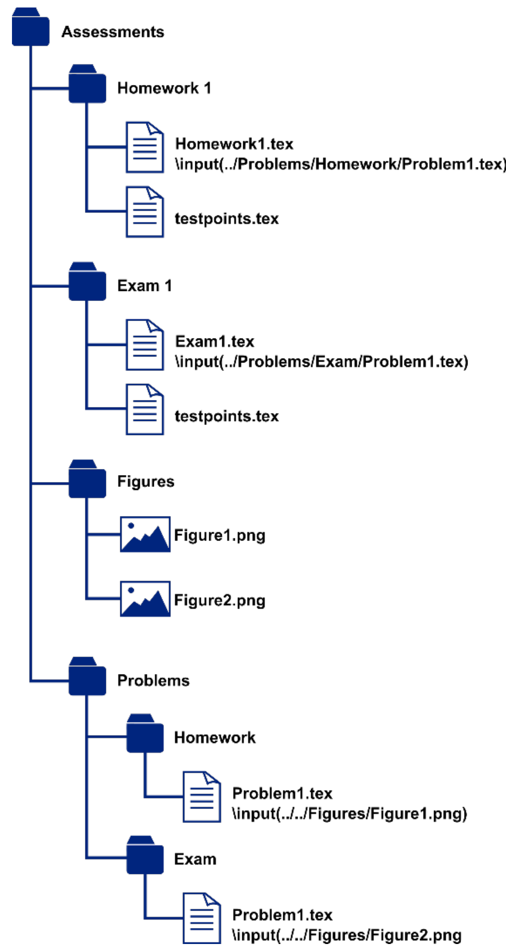

*Figure 1: An example of SLAMS file structure.*

The most powerful aspect of SLAMS is the use of the \toggle command provided by the etoolbox package. This command allows the use of Boolean logic within the tex documents. This allows the user the ability to tell LaTeX to not typeset certain file contents. In this framework, a logical variable is defined called "Solution." When the value of "Solution" is false, only the problem statement is typeset. When "Solution" is set to true, both the problem statement and solution are typeset, Figure 2. This means that a single file can contain one or more problem statements along with the corresponding solutions, improving maintainability of problem and solution sets. The following code excerpt demonstrates the usage of the toggle command. To typeset the solution, simply change the value of "Solution" to true.

```
⋮
\usepackage{etoolbox} # Package that contains the toggle command.
```

```
⋮
\providetoggle{Solution} # Define the Solution variable.
\settoggle{Solution}{false} # Set the value of the Solution
variable.
⋮
\input{testpoints}

\begin{document}
⋮
\iftoggle{Solution}{
  # Insert solution here
⋮
}
```

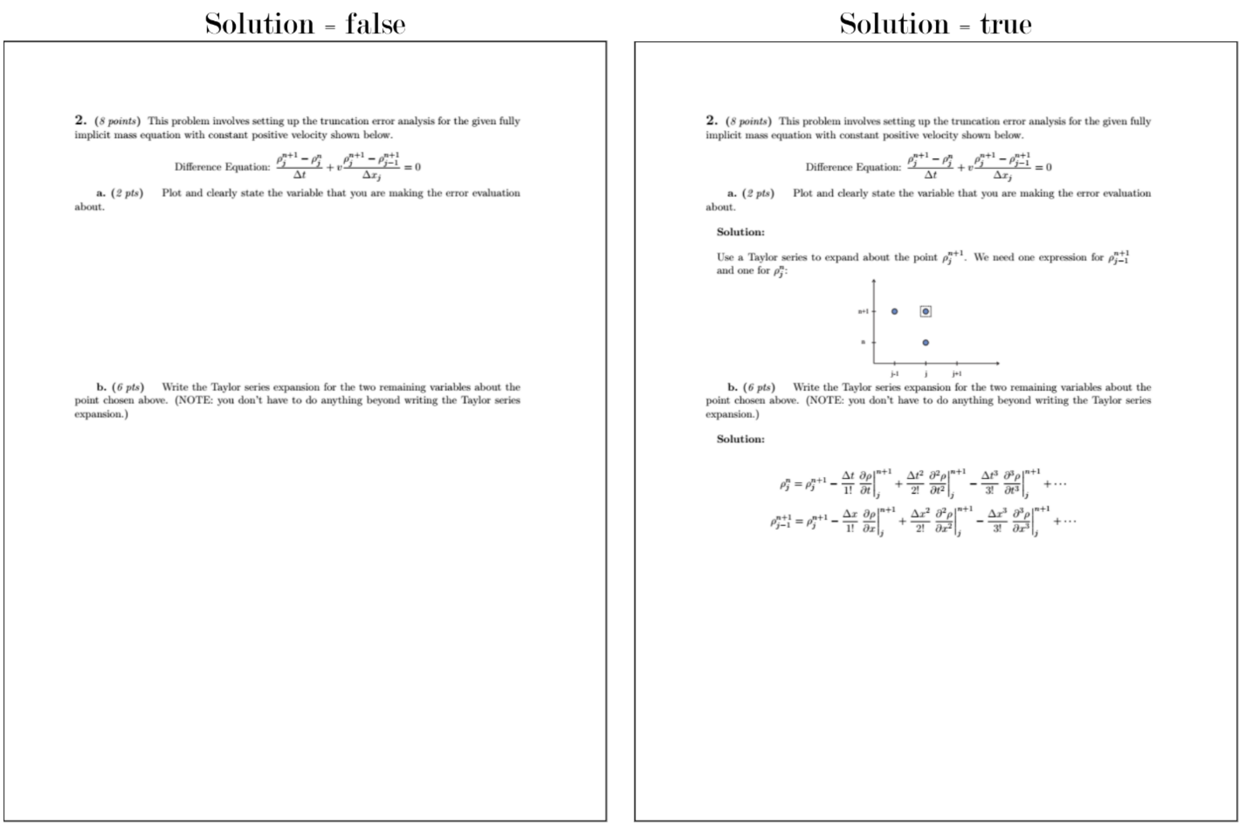


*Figure 2: Difference in typesetting when changing the solution logical variable.*

Another feature of the system is the ability to format the document according to a style template. Instructors often begin by modifying a previous assessment as a starting point. However, this approach has limitations because WYSIWYG word processors combine content and style, resulting in inconsistent document formatting. Issues like varying font types and sizes, indentations, and image sizes can persist throughout the document. Time spent fixing stylerelated problems could instead be spent improving the actual content (i.e., problems and solutions). The template can be modified to readily meet any requirements assessments set by a user's institution, such as font size, font type, margins, etc.

SLAMS is a modular and flexible framework. The method recommend in the SLAMS framework is to have a database of individual problems that can be individually included in any assessment by just pointing the assessment file to the problem using the \input command. The user can quickly switch problems associated with each assessment. Problems can be organized by assessment type, point total, topic, or any other category the user likes.

**Individual Assessment Creation**

We describe our assessment creation workflow from the instructor's perspective. The starting point for the workflow is the main .tex file which contains most of the information for the assessment except the actual problems. The authors have found it helpful to have a different main file as a starting point for homework assignment versus exams. The first section of the main file sets various document parameters and loads packages. In most cases, users do not need to modify this section. The second section includes the heading, instructions for students, honor code statement, and any other content that isn't an actual problem on the assessment.

The last section of the main file uses the \input command to insert different problems from the user's problem and solution library. As discussed earlier, the instructor can change whether solutions appear in the typeset PDF file by changing the Solution flag from false to true.

The problem file contains a problem statement in the first section and an optional solution in the second section. Both the main file and problem file can use all features of LaTeX, such as graphics, tables, equations, changing fonts, etc.

Creating the student assessment is simply performed by typesetting the main file. The main file and any included files must be free of errors for the output PDF to be generated.

**Student Feedback**

An online survey of undergraduate and graduate nuclear engineering students explored preferences for student learning assessments and solutions. Although the limited number of responses prevents robust statistical analysis, the results inform potential future research. Students unanimously preferred assessments with consistent formatting and clearly stated point values for each part of the problems. The majority preferred receiving typed solutions from instructors for assessments such as quizzes, homework, and exams rather than handwritten ones.

In contrast, preferences for completing assignments varied; some students preferred typing, while others favored handwriting on paper or digital tablets. The primary concern was the time-consuming nature of typing equations, suggesting a need for further research on whether training in typing equations could reduce this issue. A minority reported that handwriting submissions improved their learning.

**Instructor Usage**

Instructors with programming backgrounds can readily adapt the system to their specific needs. For those less familiar with programming, online LaTeX editors like Overleaf can ease the transition. The provided documentation is also expected to further streamline the integration of this system into instructor workflows. All .tex files, documentation, and sample PDF output files are accessible via GitHub. To obtain your copy of SLAMS go to:

https://github.com/jkw104/SLAMS
git@github.com:jkw104/SLAMS.git
gh repo clone jkw104/SLAMS

The authors have found the problem library to be helpful as the problems are readily available for any class. Creating an assessment now involves simply selecting the type and number of problems rather than searching through old Word documents.

**Conclusions and Future Plans**

Future enhancements are planned in two phases. The short-term goal is to develop a template for multiple-choice questions. The long-term goal is to enable generating problems with unique values for each student, promoting collaborative learning and reducing plagiarism. The unique values would encourage students to understand the concepts rather than simply exchanging answers.

**References**


1. J. J. Epstein and F. L. Van Voorhis, "More than Minutes: Teachers' Roles in Designing Homework," *Ed. Psych.,* vol. 36, no. 3, pp. 181-193. Jun. 2010. DOI: 10.1207/S15326985EP3603_4.
2. K. Santoro and R. Bilisoly, "Creating, Automating, and Assessing Online Homework in Introductory Statistics and Mathematics Classes," in *Joint Statistical Meeting,* 2014, pp. 1787-1798. Available: https://arxiv.org/ftp/arxiv/papers/1501/1501.03215.pdf
3. D. G. Whiting and D. T. Scott, "YSTATTEST: A System for Automated Online Test Creation and Correction," in *Intl. Conf. on Teaching Statistics,* 2006, Available: https://iase-web.org/documents/papers/icots7/C135.pdf
4. L. Taalman, "A Gentle Introduction to Tex," Duke Mathematics Department. https://sites.math.duke.edu/computing/tex/latex.html (accessed February 5, 2024).