A Computer Based Approach For Generating Standardized Student Assessment Using A First Year Biology Program As A Model To Evaluate Its Effectiveness

Todd Nickle [*] ***and Jeff Sheedy***

Department of Chemical, Biological and Environmental Sciences
Mount Royal College
4825 Richard Road S.W.
Calgary, AB, Canada, T3E 6K6
[*]Address correspondence to Todd Nickle
403-240-6682
tnickle@mtroyal.ab.ca

Jeff Sheedy
403-240-1182
jsheedy@mtroyal.ab.ca

Todd Nickle is a native Calgarian who had the luck to return to his home city after getting his PhD at Oklahoma State University. When not teaching Biology, he chases his young son and daughter in a vain attempt to keep them out of mischief.

Jeff Sheedy obtained his BSc in Agriculture from the University of Alberta and after spending a year in Australia, he joined the Faculty of Science and Technology at Mount Royal where he has been an Instructional Assistant for 20 years. Jeff manages the Biology Department Greenhouse along with various other Biology lab duties and as a diversion, he coaches the Mount Royal College cross-country running team.

Reprinted From: Nickle, T. and J. Sheedy. 2000. A computer based approach for generating standardized student assessment using a first year biology program as a model to evaluate its effectiveness. Pages 424-427, *in* Tested studies for laboratory teaching, Volume 22 (S. J. Karcher, Editor). Proceedings of the 22nd Workshop/Conference of the Association for Biology Laboratory Education (ABLE), 489 pages.

Abstract

Biological Sciences education at Mount Royal College accommodates students from a variety of backgrounds to fulfil transfer requirements to complete a degree at the University level. We possess a large undergraduate Introduction to Biology program which requires coordination of several instructors to administer laboratory material to many small classes of students. To ensure consistency of student skill evaluation and to reduce the workload on instructors, we have designed a system for creating quizzes that can be presented to students at the beginning of each laboratory. In this workshop, we will demonstrate the software we designed which generates quizzes and answer keys on demand. The nature of the software and the logic of question design enables the program to handle "data cassettes" which can be transferred across a variety of computer platforms and which are relevant to any discipline for which multiple choice assessment is possible. The workshop will conclude with our observations of how the software has helped lab coordination, areas that need further investigation, and future enhancements to our protocol.

Background

The Introductory Biology class taught at Mount Royal College serves well over 800 students per year. Because of this, we have enlisted many different instructors to supervise and manage the labs. Because they all have different skills at designing questions, we decided to centralize all our quiz questions and select random questions from the common pool to distribute the questions to the students. Students had expressed concerns about differences between individual instructor demands, and faculty identified consistent biases regarding the mean laboratory grades that corresponded to the instructors teaching each lab. We implemented weekly quizzes to ensure students had read and understood the laboratory concepts.

Because laboratories run all week long, we wanted to have a number of quizzes to choose from so there would not be an advantage to students taking the quizzes later who would already have heard about the quiz contents. Many of our instructors have additional jobs off-campus, and so we wanted the questions to be available remotely. Making the files available through a Web Browser would have the additional advantage of allowing Macintosh, PC, or even Unix computers to be used to construct the files. Our final concern was that a standardized marking scheme be implemented to minimize individual instructor biases.

This paper describes the structure of the program that generates the quizzes (algorithm), demonstrates how to generate quizzes from the Web server, and also describes how new question files and questions can be generated.

Algorithm

The program was designed to display multiple choice questions with an answer key. It also includes short answer response questions (without a key). The questions are extracted randomly from a file containing numerous questions. The questions are then displayed either as a print-ready format (to be run off for students) or an interactive Web version that displays the answers after the form is filled out by the student.

The multiple-choice questions are designed in a special way to increase the amount of variability. The basic structure of a multiple-choice question is shown in Figure 1.

Statement:
A. False answer
B. Another false answer
C. True answer
D. Yet another false answer

Figure 1: Structure of a multiple choice question

It's often the case that several different "true" answers could apply equally to a particular statement. The program takes advantage of this by randomly selecting a true answer from a list, randomly determining where in the question it should be placed, and then padding the rest of the answers with false answers. By entering several "true" answers and several "false" answers into the database, an incredible number of variations can be obtained for EACH question. This provided us with sufficient variability to ensure no overall advantage to any of the laboratory sections. Only one file of questions is accessed at any one time, and questions are grouped together (in my examples, "Lab 1" questions are in a single file and deal with microscopy, "Lab 2" questions are pooled and deal with macromolecules.

Implementation

Instructors were invited to submit questions for the database. Each question was submitted with: 1) A statement; 2) One or more answers that were "true" for that statement; and 3) Three or more answers that were "false" for the statement. These were formatted for the program and saved in a directory accessible to the PERL (a programming language) code file which created the quiz. New quizzes were generated just prior to the laboratory in which they were to be written.

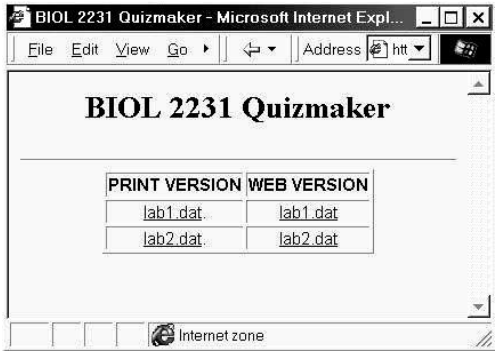
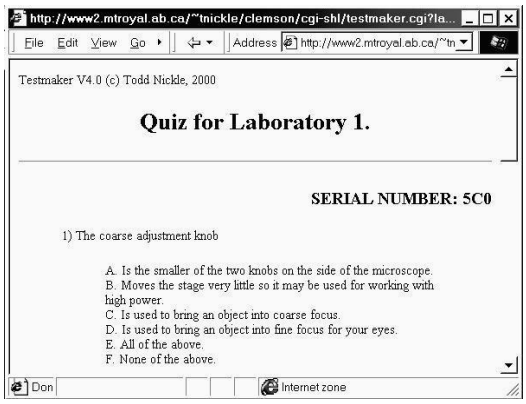


Figure 2: Selection of quiz data and format

Creating Questions

Quizzes based on the database can be accessed by navigating to the website introduction page (for an example, log on to http://www2.mtroyal.ab.ca/clemson/make_quiz.cgi). Data files are identified by their names ending with ".dat" and these are displayed as either "Print" or "Web" formats (Figure 2).

Quizzes formatted as a "Print Version" have an answer key at the bottom and a unique random serial number at the top (Figure 3). The serial number serves to help match student versions of the quiz with the appropriate answer key.



The Web version of the quiz contains "radio buttons" which restrict answer choices to one selection for each question (Figure 4). The web version allows students to access the quiz base from home to help them hone their skills and identify weak areas.

CREATING QUESTIONS

Questions can be added using an online form. The first step is to decide what file you wish the quiz questions to be appended to. If you want to start a new file, this can be done as well. All data files MUST end with ".dat" or the server won't show them in the listing. You can try out the protocol for adding new questions at the site http://www2.mtroyal.ab.ca/~tnickle/clemson/add_data.cgi.

The first step in choosing the data file is to select either an existing file (listed at the bottom of the form) or enter a new, unique filename (ending with ".dat") in the space provided (Figure 5). Clicking on an existing file immediately takes you to the form for entering the question (Figure 6). For each new question, enter a statement in the large text box. For each of the text boxes following, enter a response and click in the radio button to the left of each response whether it relates "true" or "false" to the statement. Submit when done.

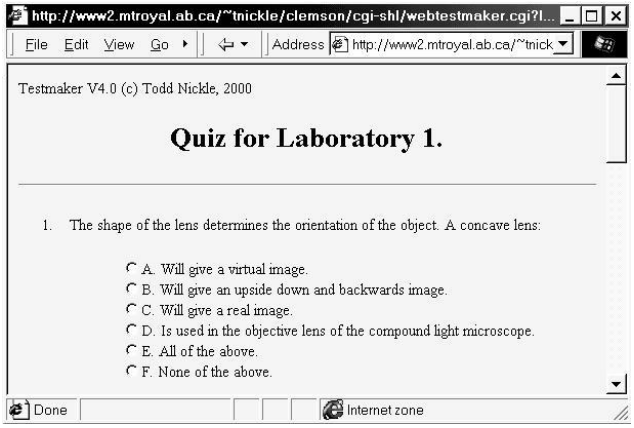


Figure 4: Interactive (web) version

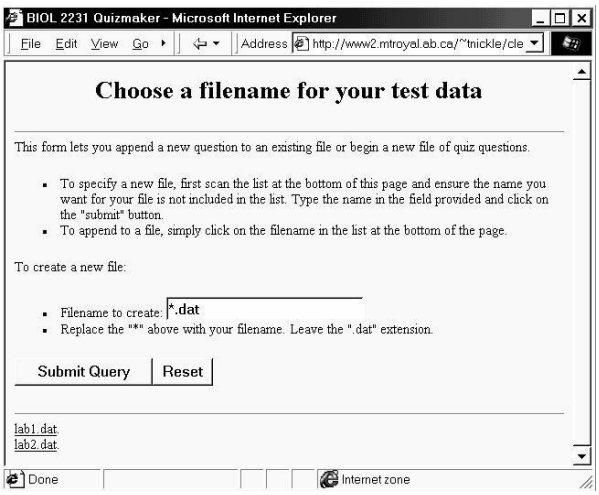


Figure 5: choosing a filename to add data to. If it's a new file, replace the "*.dat" in the text box with the new filename (ensure the new name ends with ".dat"). You can append to an existing file (e.g. lab1.dat, bottom left).

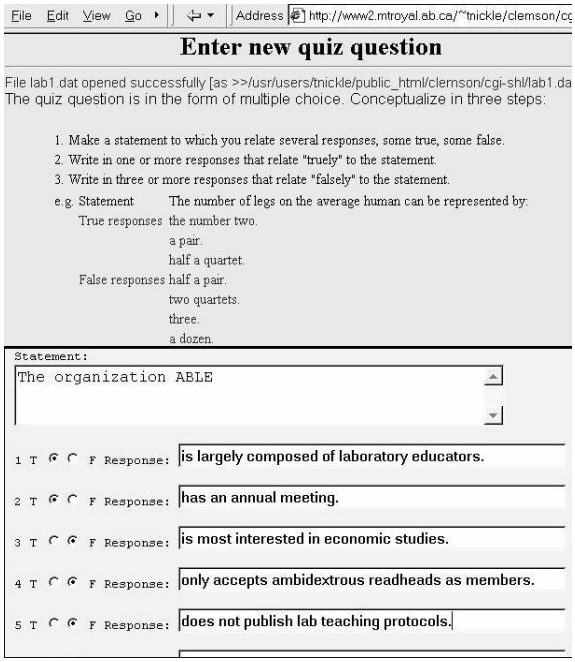


Figure 6: Entering a new question.

Conclusion

The introduction of the online quiz bank streamlined the process of generating new, unique quizzes for the students. Instructors could create one or more quizzes within one minute and print them to a transparency so the students could take them during the laboratory period. Previous concerns about the time it took to create a quiz and individual instructor biases were eliminated with this system.

Since the implementation of this system last year, we have shifted the marks normally allocated for the quizzes to a major laboratory project and eliminated the quizzes. The effort has not been without benefits, however. The web-based interactive version of the quiz was made available to students and has been of benefit in allowing them to self-assess their understanding of the topics. In addition, a small portion of the midterm examination is based on the questions in the database (this allows us to set up an extra station where they can answer questions while awaiting one of the practical stations to be vacated by other students).

The software demonstrated at the Clemson ABLE meeting is accessible at http://www2.mtroyal.ab.ca/~tnickle/clemson. Feel free to add questions and create new files. If you are interested in the PERL code that runs the system or would like to have a tailored version constructed, contact Todd Nickle.