OPTIMIZATION OF STUDENTS' STUDY HABITS USING ON-LINE TESTING

Eva Milková

Faculty of Informatics and Management, University of Hradec Králové
Rokitanského 62, Hradec Králové, Czech Republic

Keywords: On-line testing, virtual learning environment, multimedia study material.

Abstract: Multimedia applications give us an excellent chance how to support not only demonstrating and visualizing the explained subject matter to be much clearer and comprehensible, but also enable us prepare various kinds of test for students. The feedback provided through tests helps students to recognize what they already knew and what they didn't know and need to improve. In the article we introduce several on-line testing applications that we use for our students and will discus their educational potential.

1 INTRODUCTION

Virtual learning environment and multimedia applications give the teachers big chance to use them to optimize students' study habits, which means to make their study more effective, time-efficient and explained topics more comprehensible.

Last year, in the paper prepared for WEBIST 2007 (Milková and Slabý, 2005) we summarized the biggest advantages of the virtual study environment WebCT used as a compliment to the teaching of subjects taught within the full-time study programs at our faculty. Let us remind them now as well:

- Students have all electronic study materials assigned to the appropriate subject available altogether in one environment together with needed information.
- The relevant study material is accessible whenever and wherever. In this way also students staying abroad within the framework of the Socrates Erasmus students exchange for several months are able to follow the subject.
- When lessons are taught by more than one teacher, all teachers can be included as teaching assistants to the e-subject. Each of them inserts settings of exercises and the evaluation of credit tests for his/her students to the WebCT environment. Thus the lecturer (guarantor of the course and author of the e-subject) has a clear survey of both the history of lessons and results of all students.

- Students interested in the area explained within a subject can find additional material in the WebCT environment, and sources and information outside the immediate framework of the subject. Discussions on "a level" with their teacher can start. This is also one of the ways in which the teacher can train his potential thesis writers.
- In many subjects, the inevitable outcome of creating common projects is teamwork. Virtual study environments are inherently well-suited to this. Students can communicate with the teacher and each other, and discussions are accessible to members of the team only. Even as geographically isolated members of a virtual team, they can send requests for advice and parts of project to the teacher for oversight.
- Students greatly appreciate accessibility to self-tests with automatic checking and to multimedia study materials visualizing the explained matter.
- Students find also very useful the option called Calendar, where they can see all the information about important dates (written credit tests, examination etc) in one environment.

In this paper we devote attention to one of the-above mentioned advantages, namely to the accessibility to self-tests with automatic checking prepared in WebCT and to the other multimedia

applications supporting self-preparation of students through on-line testing. "The feedback provided through tests helped students to recognize what they needed to know, appreciate what they already knew and understand, what they needed to do in order to learn, and what they didn't know" (Black and William, 1998).

2 ON-LINE TESTING IN VIRTUAL LEARNING ENVIRONMENT WEBCT

WebCT, like all the other similar virtual learning environments, offers tools enabling preparing various kinds of tests with several types of test-questions.

2.1 Three kinds of Tests in the WebCT Environment

There are three kinds of on-line tests in the WebCT environment in the option called Assessments: *Quizzes*, *Self tests* and *Surveys*.

The following text is a copy of the text written in http://oliva.uhk.cz/webct/help.dowebct.

- A quiz is an online test that students complete and submit for marks. The marks are recorded in Grade Book. You can use quizzes to assess students' performance in the course.
- A survey is an online questionnaire that students complete and submit anonymously. A survey is not worth marks but will be recorded as Complete in Grade Book after it is submitted. You can use surveys to allow students to give feedback or opinions.
- A self test is an online test that students complete and submit for marks so they can assess their understanding of course material. The marks are not recorded in Grade Book.

Remark: Grade Book is the WebCT tool that serves to the teacher to view, enter, and manage grades for students.

2.2 WebCT Test-questions

In the classical (non electronic) tests prepared to test knowledge in natural science subjects there have been used two main types of questions for years (centuries); the type *Calculated* and the type *Multiple Choice*. However, in the textbooks directed to the foreign languages education there have been used several other types of questions. These types are available in the WebCT environment as well. Let us introduce their list and short description.

- *Calculated*: Students are presented with a mathematical formula and must enter the answer.
- *Combination Multiple Choice*: Students are first presented with a list of answers. They are then presented with various combinations of those answers and must select one of the combinations as correct.
- *Fill in the Blank*: Students are presented with a question containing blanks and must provide the missing text.
- *Jumbled Sentence*: Students are presented with a question containing blanks for which they must provide the missing text by selected a word or phrase from drop-down lists.
- *Matching*: Students are presented with two lists and must match terms in one list with terms or definitions in the other list.
- *Multiple Choice*: Students are presented with a list of answers and must select one or more answers as correct.
- *Paragraph*: Students must answer the question by entering multiple sentences.
- *Short Answer*: Students must answer the question by entering a brief response.
- *True False*: Student must select whether the statement is true or false.

All types of questions, except the type Paragraph, can be checked automatically. Thus students can see correct answers immediately after sending solutions of all questions (except the type Paragraph).

2.3 A Case Study: On-line Testing in the WebCT Environment Prepared for the Subject Graph Theory

As we have already mentioned in natural science subjects above all the types Calculated and Multiple Choice have been used. However, the other types of question seemed to give a chance to solve tasks in an enjoyable way. Therefore we tried to prepare for the subject Graph Theory test-questions used the other types as well. We were successful in it and some examples are shown in the following figures 1 - 5.

It is very important that the WebCT environment enables to include figures and tables within the questions. Without this possibility it would be unthought to create questions for the subject Graph Theory based on images of graphs.

There is also the possibility to change the HTML code of questions in the WebCT environment. We have used this opportunity when completed some questions with the area for writing some notes. Students often need to record some information when solving the task. Therefore we have increased the appropriate questions by "area for notes" prepared in very easy way using HTML tag <TEXTAREA> </TEXTAREA>.
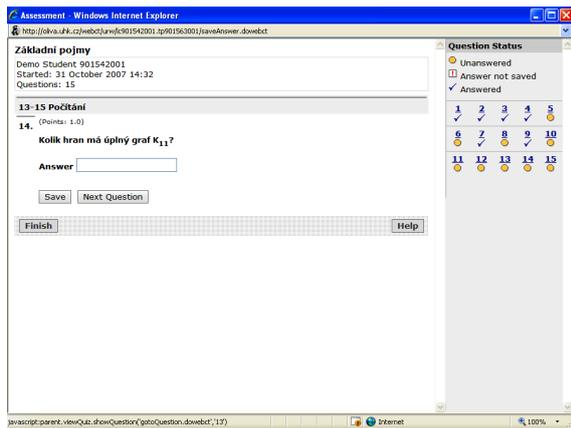
Figure 1: A question of the type *Calculated*; "How many edges there are in the complete graph K_{11}?".
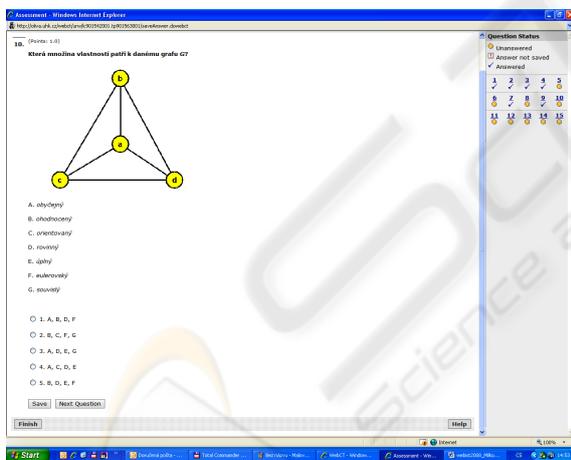
Figure 2: A question of the type *Combination Multiple Choice*; "Which set of properties written in A. to G. belongs to the given graph?" (The properties written in English: A. undirected, B. weighted, C. directed, D. planar, E. complete, F. eulerian, G. connected. Thus the only 3rd combination A,D,E,G is correct).
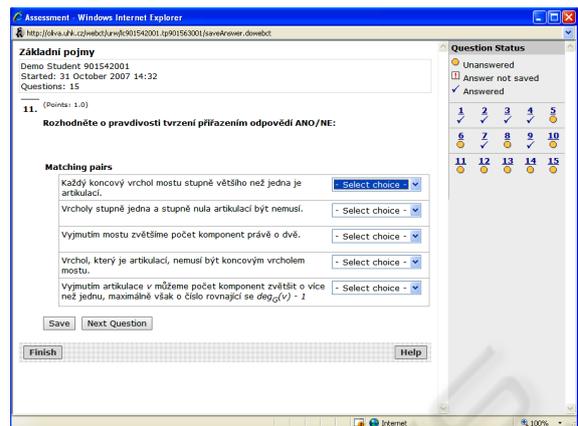
Figure 3: A question of the type *Matching*; "Determine the truth of each of the following statements using the either answer YES or NO." (There are several short statements and about each statement students should decide if it is true of false statement. As e.g. the first statement: "Each end-vertex of a bridge with the degree bigger than one is a cut vertex." This statement is true thus students should select the answer YES).
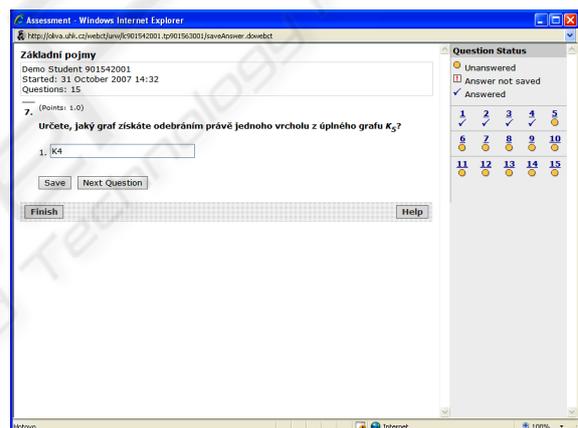
Figure 4: A question of the type *Short Answer*; "Determine the graph that you gain by removing one vertex from the graph K_5?".

We are pleased to say that students like these types of questions that can be solved in an enjoyable way. We do not use on-line testing as a part of examinations. The prepared tests serve for students to recognize themselves what knowledge they need still to improve. Students can, but don't have to solve them during the term. However, similar tasks occur in the exam test. It is up to the students whether they deal with these questions and if they do not understand them well they can discuss them with their teacher during the lessons.
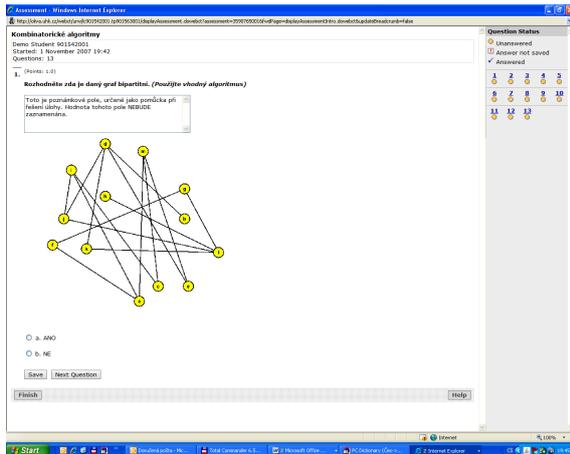
Figure 5: A question of the type *True False* where the "area for notes" is used; "Determine if the given graph is bipartite or not. (Use an appropriate algorithm.)" (Students record list of vertices and edges in order given by the used algorithm Breadth-First-Search and looking for the circle of odd length. If there is no such a circle their answer should be YES. In the opposite case their answer should be NO).

3 INTERESTING WAYS OF ON-LINE TESTING

In the previous chapter we have described tool enabling to prepare tests in WebCT environment. Nevertheless, the types of questions prepared by WebCT tool need not be easy to fulfil for each subject as e.g. for the subject where students learn how to create algorithms.

In the following text we illustrate a nice application called ALGDS (Hotový, 2007) containing interesting types of test-questions making testing of algorithms amusing at first and then we emphasize the possibilities of the program ALGORITHMS (Voborník, 2006), the fundamental tool giving feedback to students studying the subject Algorithms and Data Structures.

However, to appreciate advantages of the above-mentioned tools it is necessary, at least briefly, explain the syllabus of the subject Algorithms and Data Structures.

3.1 The subject Algorithms and Data Structures

The ability to create the basic algorithms contributes to a development of logic consideration and imagination and is an inseparable part of students' skills studying Application Informatics and Information Management at our faculty. And because the entrance examination at the faculty doesn't include questions concerning the above-mentioned area there are a lot of students without any algorithmic knowledge at the beginning of their studies. Therefore we devote the whole first term for increasing the algorithmic thinking of our students in the subject called Algorithms and Data Structures (ALGDS). This subject is placed before the other subjects dealing with algorithmic and programming skills.

The education in ALGDS lacks all the structures and constructions connected with structured paradigm. Consequently students do not get any habits which might make the entrance into object oriented world more complicated. (Remark: The entire conception of programming education at our university can be found in (Kozel, 2006)). Creation of algorithms is based on the imagination of a brick-box, a nice and useful game for children. There are only several base elements available from which children are able to create incredible buildings. Why do not use this idea also when creating algorithms?

Thus when we lead our students' first steps by creation of algorithms we explain to them that it is like building interesting objects out of just a few basic elements. In the subject AGDS it means that we start our teaching with basic algorithmic structures (basic elements from the brick-box) and typical algorithmic structures (a few parts made out of these elements) and then we let students get into the secrets of making whole algorithms (building whole constructions). Algorithmic structures and algorithms are written in Czech meta-language (Czech translation of basic commands used in the program-language Pascal) on a paper!

At our lectures and lessons we devote time not only to the explanation of algorithmic structures and their illustration on several examples but we also let students complete prepared algorithms similarly as you can see in the following two examples. This is very welcomed by students and makes lectures much more interesting and enjoyable.

- Complete the algorithms solving the following task. "In the sequence of *n* integers saved in the array `a` (in items `a[1]` , ..., `a[n]`) determine the first minimum value and then sum of all integers behind the found minimum value."

```
begin
  minimum := a[1];
  sum := ..........;
  for i := 2 to n repeat
      begin
      sum := sum + ..........;
      if a[i] ... min then
        begin
          minimum := ..........;
          sum := ..........;
        end;
    end;
end.
```

- There are *n* integers saved in the array a (see the table). Determine the values in the array a after finishing the following algorithm. Write them to the table.

```
begin
  n:=6;
  x:=a[1];
  i := 2;
  while i ≤ n - 1 do
    begin
      if a[i] > x then
        begin
          a[1]:= a[i];
          a[i]:= x;
        end;
      i := i + 1;
    end;
end.
```

a[1]	a[2]	a[3]	a[4]	a[5]	a[6]
11	8	19	7	16	17

Similar tasks occur in examination tests. To be able to solve such examples we have created the application ALGDS where students practise their knowledge in the above-mentioned way. Let us briefly describe it.

3.2 Web Application ALGDS

Web application ALGDS is created within PHP language and it is intended for testing knowledge gained in the subject Algorithms and Data Structure. The link to the application is placed in the WebCT environment within the subject Algorithms and Data Structures. The design of the application is illustrated on the figure 6.

Figure 6: The first page of the application ALGDS.

Web application ALGDS *enables three types of questions*.

- Complete the given algorithm.
- Order the given algorithm.
- Solve the given algorithm.

The test *Complete the given algorithm* demands to complete omitted parts in the algorithm solving the given task similar as it was shown in the first example illustrated in the previous section 3.1.

The test *Order the given algorithm* demands to order all commands written in wrong order to get algorithm, which correctly solves the given task (see figure 7).

The test *Solve the given algorithm* demands to determine values of some variables used in the algorithm using the given enter data similar as it was shown in the second example illustrated in the previous section 3.1.

```
end.
```
```
while (i <= n) and (found = 0) do
begin
```
```
i := i + 1;
end;
```
```
if (a[i] mod 7) = 0 then
begin
```
```
write(a[i]);
found := 1;
end;
```
```
begin
found := 0;
i := 1;
```

Figure 7: A question of the type *Order the given algorithm* used in the application ALGDS; Arrange mixed up parts of the given algorithm in correct order to get algorithm finding the first value divided by 7 among values saved in the array a.

302

The Web application ALGDS enables students to improve their knowledge in an enjoyable way. Each students answer is analyzed and the correct answer appears if student was wrong. There is no explanation of why the student's answer was false however they can follow step by step the whole algorithm using the program ALGORITHMS.

3.3 Program ALGORITHMS

We have already mentioned that students write their algorithms on paper. There was an important question. How can students get feedback for their solution written on papers in the Czech meta-language when studying at home and preparing for lessons? They have two books (Milková, 2001), (Milková, 2004) containing many solved examples, prepared for the subject Algorithms and Data Structures, available. Nevertheless, there are a lot of other tasks that we give our students to solve and therefore the program ALGORITHMS was developed. This program is created in Borland Delphi environment within the thesis of our student on the script given by author of this paper. The program is user friendly and provides entire graphical comfort for users.

Using the program ALGORITHMS students can place their solutions of the given tasks in the program and the program *shows them step by step how their algorithms work*, if they are correct or not. The program also shows *actual values of used variables* in each step of the algorithm's process. In this way students can test their knowledge gained in the subject Algorithms and Data Structures.

4 CONCLUSIONS

In the paper we illustrated some of on-line testing possibilities, various kinds of feedback that helps students to recognize what they already knew and what they didn't know and need to improve.

Let us emphasize once more that we do not use on-line testing as a part of examinations. The prepared tests serve for self-preparation of students to be able to recognize what knowledge they need still to improve. Students can, but don't have to solve the prepared test. However, similar tasks occur in the exam test. After more than tree years experience with this approach to the increasing the students' preparation to a subject through on-line testing that offer immediate response we can say that most students deal with tests and if they do not understand the questions well they discuss them with

their teacher during the lessons which serve to the teacher as an important feedback as well.

REFERENCES

Milkova, E., Slabý, A., 2007. Virtual learning environment: Challenge to Increase Teaching and Learning. In: *Third International Conference on WEB Information Systems and Technologies (WEBIST 2007)*. INSTICC. Barcelona.

Black, P., William, D., 1998. Inside the black box. *Phi Delta Kappan*, Vol 20, No.2, pp 139-148.

Hotový, V. 2007. *Vizuální podpora rozvoje algoritmic-kého myšlení s využitím webdesignu*, Thesis. Hradec Králové.

Voborník, P., 2006. *Programovací jazyk pro podporu výuky algoritmů*, Thesis. Hradec Králové.

Kozel, T., 2006. Java ve výuce programování. In: *Proceedings of Informatika XVIII/2006*. Konvoj. Brno.

Milková, E., 2001. *Algoritmy – typové konstrukce a příklady*, Gaudeamus. Hradec Králové.

Milková, E., 2004. *Algoritmy v příkladech*, Hradec Králové. Gaudeamus.