TESTING THE PROGRAMMING KNOWLEDGE AND SKILLS IN ELECTRONIC ENVIRONMENT

Martin Cápay
Constantine the Philosopher University in Nitra, Slovak Republic
E-mail: mcapay@ukf.sk

Abstract

Using of modern concepts based on information and communication technologies is still growing and also educational organizations must react on this situation. It requires effective approach in control, teaching and testing process. Classification usually follow-up to testing process is very sensitive phase of the learning process and it is necessary to ensure the greatest possible fairness and objectivity. Assessment of programming knowledge and skills is especially challenging task. To ensure the objectivity of the assessment, it was necessary to proceed to develop a unified system, at least in the case of testing knowledge of the theoretical part. The paper presents the model of computer-aided knowledge testing with regard to the content of programming subjects. We developed and implemented model using the standard quizzes in electronic environment Moodle. It is considered such as the most accessible environment alternative for educational organizations. The research outcomes show that the elaboration of the content into the thematic areas of the model, and specification of their objectives are reasonable. Positive perception of electronic testing is clear, but contrary the students prefer combine form of testing, where teacher still plays the essential role.

Key words: *automated assessment, electronic test, on-line testing, programming, LMS Moodle.*

Introduction

Assessment and classification is very sensitive phase of the learning process and it is necessary to ensure the greatest possible objectivity. Very important nowadays factors are effective on time and used sources, which might be reaching by didactical test in electronic environment. In some science disciplines, e.g. mathematics, using mass didactical tests as normalize tools of objective measurement is practicable. In case of computer science, especially in programming subjects, application of didactical test is more difficult. If we concentrate on theoretical knowledge (for example language syntax, instruction, common principles), practical skills stays unverified. On contrary, we could test only practical skills without its theoretical background. This is commonly better way, but time factor is very important here. Every student needs to work with own computer station for longer time, but capacity of testing rooms is limited and usually unsatisfactory. Tutor has to get more examine terms. Other programming skills testing disadvantage is subjectivity in quality of solution evaluation. Speaking from experience, different assignment demands distinct programming skills and different tutor give distinct importance to another partial solution.

We interested in application of such way of testing, which at least reducing the degree of subjectivity in testing programming, but we do not want to ask a question which require rote memory for a correct response. Perhaps, because questions that require recitation of facts take less time, teachers sometimes avoid asking higher-level questions (Kipper & Rüütmann). However, there was not a database of higher-level questions and tasks that would require an active application of knowledge and programming skills for correct answer, and in which the correctness of responses could be evaluated automatically

Methodology of Research

In the initial phase of research we were looking for a system in which later created question bank could be implemented. The aim was to use only simple standard questions types, supported by most of electronic systems, so we should ensure their later portability between any environments. We also required the on-line data and statistical processing, automated control, evaluation and assessment.

We could summarize, that exist three types of system:

- System built directly for specific subject – it consists from learning activities and question bank. Changes and edits are forbidden.
- Copyright system – teacher might specify learning activities, bank of question etc. Changes and system edits are forbidden.
- Empty system – teacher must specified learning activities, bank of question etc. Changes and system edits are permitted.

After web pages analysis of many institution as well as contributions from the journals and conferences we found, that exist quite enough copyright systems for testing. Here are a few of them (Cápay, 2010a): WBPES (Web-Based Public Examination System, Dhaka University, Bangladesh), QUIZIT (Chemware Ltd, New Zeeland), ASSYST (ASsesment SYSTem, University of Liverpool), TRAKLA2 (Helsinky University of Technology), PILOT (Platform-independed Online Tools, Johs Hopkins and Brown university), WOES (The Web based On-line Examination System), VEVAL – Virtual EVALuation (Alpen Adria University, Klagenfurt, Austria). From 1994 also Slovak universities have try to create automated test system, e.g. UnivTest, Tester, UDABTAM, Avytez (Cápay, 2009b). Of course there are a lot of other universities and institutions which carried out their own copyright projects, successfully used only within the university. Each founded system was created based on pre-specified requirements. Therefore they are usually suitable only for a narrow or specific field of application.

After analysis of copyright system our attention turned to the management and learning management systems. The analysis show that companies in Slovakia (usually banks) prefer environment adjust to their company requirements. Most of them have implemented iTutor, which is solution with commercial support. Universities in Slovak and Czech use a wide range of commercial or open learning management systems such as Claroline, Fle3, ILIAS, MS Class Server, WebCT, Eden, Enterprise Knowledge Platform, LearningSpace, eAmos, eDoceo, Uniforms, uLern, Aspen, Oracle iLearnin, NETOPIL School and Moodle (Cápay, 2009b). On contrary to companies, universities usually chose empty open system Moodle, which is increasingly appearing also in secondary schools.

LMS Moodle

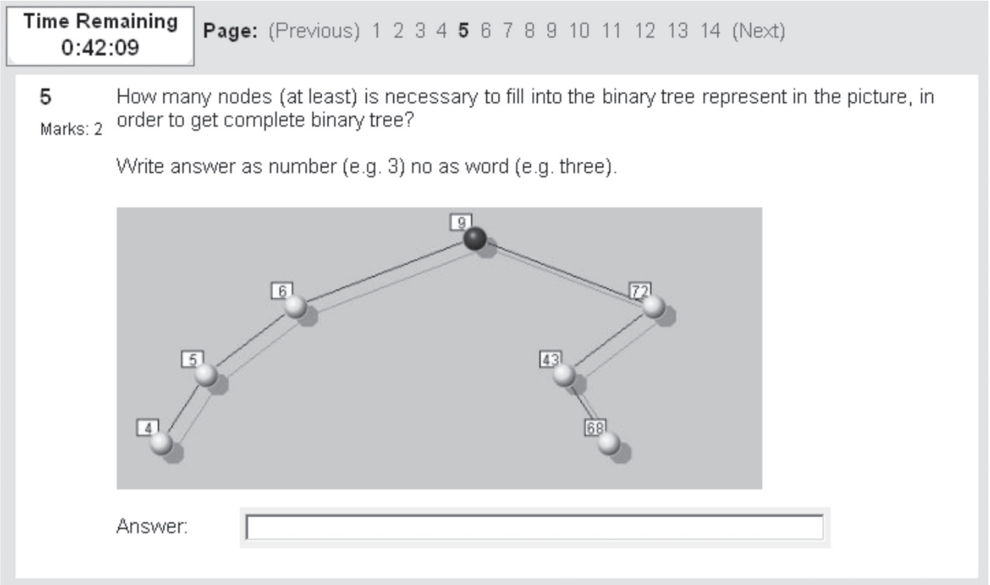
The main outcome of our analysis is, that each LMS systems have implemented testing module. Each LMS support the standard type of question e.g. multiple choice, true/false, short answer and sometimes matching. It can be considered as a good solution for universities to implement the testing process in any of existing LMS systems. Taking place and time into consideration, we can state a several type of teaching (Glusac et al., 2007):

- Same time, same place.
- Same time, different place.
- Different time, same place.
- Different time, different place.

Our department have decided to use Moodle such as e-learning support environment for face-to-face study program, combining "same time, same place" and "different time, different place" approach. It is currently utilized in most of taught subjects. We have trying to find the optimal way of combining traditional chalk & chalk methods used in classroom together with ICT approach typical for e-learning in order to enhance the quality and efficiency of programming teaching and programming learning (Palmárová , 2008).

Using Moodle is recommended not only because of cost (free open software), but mainly because it is a system that has a growing user community and its possibilities are continuously enhanced. This LMS provides a quick statistical report and item analysis of tests, with the possibility of export to different file types. There are two types of item analysis, quantitative and qualitative, and is used for exploring the characteristics of single items of the test. It is suitable for judging of each of its questions (Kučera et. al, 2008). Information about all students' activities is automatically kept, thus teacher has a powerful tool for the analysis of his/her "work" during the semester. We can say that Moodle offer interactive learning portfolio of a student and the methodical portfolio of a teacher, both saved electronically within the online course (Palmárová & Lovásová, 2008).

Upon mentioned facts we have been decided to implement model of testing using standard Moodle activity Quiz. It complies with all requirements and during several years of usage we have verified its stability.



Time Remaining
0:42:09

5
Marks: 2

How many nodes (at least) is necessary to fill into the binary tree represent in the picture, in order to get complete binary tree?

Write answer as number (e.g. 3) no as word (e.g. three).

Answer:

Figure 1. Environment of Moodle modul Quiz during Programming 2 testing, category Binary tree terminology.

Distribution of Programming Content into the Categories of Test Questions

The introductory programming course is separated into two subjects, Programming 1 and Programming 2 (two semesters). The first semester curriculum contains fundamentals of structured programming in Borland Delphi. Key terms are “data types, variables and constants, operators, looping, arrays, files, functions and procedures”. The second semester curriculum contains enhance/advance terms and techniques (recursion, backtracking, pointers, linked lists, binary trees, sorting and searching algorithm).

We developed and implemented model of computer-aided testing with regard to the mentioned content of programming subjects. There was 28 thematic areas specified (table 1, detailed described in Cápay, 2009b & Cápay, 2010b) and built up a database of 237 questions and tasks appropriate for testing knowledge of the subjects of programming formulated for implementation into the selected computer system.

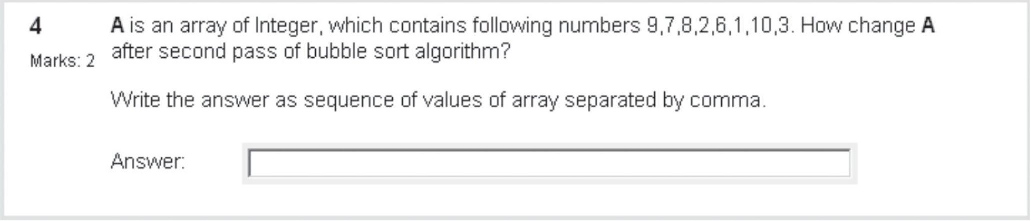
Question categories were created according to a predetermined goal of testing and assumptions to master them (see table 2). Questions in one category were considered as homogeneous (each question could be replaced by another from the same category). All questions in the same category have same score, so it is possible to choose them randomly to the test by category.

Table 1. Names of topics (categories) of test questions bank.

Categories – Programming 1		Categories – Programming 2	
1.	Algorithm properties	1.	Numbers of recursions call
2.	Flow diagrams language	2.	Recursion subroutine result
3.	Variable identifier	3.	Dynamic memory allocation and pointer
4.	Assignment statement	4.	Dynamic data structure
5.	Logic term notation	5.	Dynamic variable and statements
6.	Result of notation	6.	Searching of linked list
7.	Function and procedures of String	7.	Linked list modification
8.	String variable and loop	8.	LIFO and FIFO
9.	Data flow in different loop types	9.	Binary tree terminology
10.	Array declaration	10.	Searching of binary search tree
11.	Data type appoint	11.	Binary search tree
12.	Assignment of two dimension array	12.	Sort algorithm definition
13.	Subroutine call	13.	Application of sort algorithms
14.	Formal parameters		
15.	Procedures and functions		

Table 2. Specification of category Searching of binary search trees.

Searching of binary search trees	
Goal	Verify the ability to access to the nodes of dynamic data structure, binary tree, according to a predetermined search strategy.
Assumptions	• scanning of binary tree using recursion , • searching strategy – PreOrder, InOrder, PostOrder.
Question type	short answer
Score	2



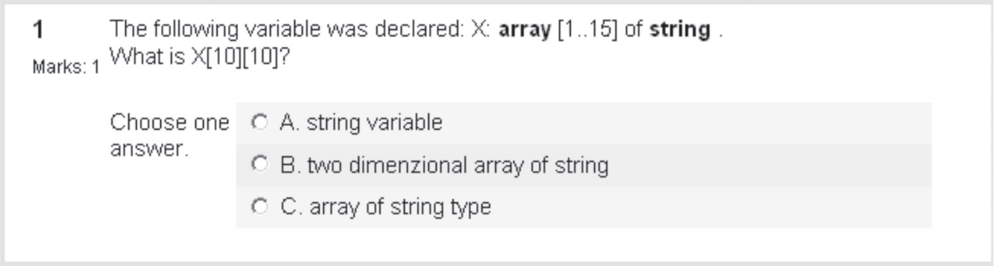
4
Marks: 2

A is an array of Integer, which contains following numbers 9,7,8,2,6,1,10,3. How change **A** after second pass of bubble sort algorithm?

Write the answer as sequence of values of array separated by comma.

Answer:

Figure 2. Presentation of close answer question, short answer type, category Application of sort algorithm.



1
Marks: 1

The following variable was declared: X: **array** [1..15] of **string** .
What is X[10][10]?

Choose one answer.
○ A. string variable
○ B. two dimenzional array of string
○ C. array of string type

Figure 3. Presentation of close answer question, multiply choice type, category Array declaration.

We decided to set following properties of test:

- random generated question from the database by categories. Each question was used a random number of times,
- mix of questions in the test,
- mix the answers in case multiple choice questions.

In the early years of using Moodle, we conducted short surveys oriented to ongoing learning at the end of the semester. We questioning about the content and form of testing in Moodle, as well as electronic testing in general.

Our model was verified in research, which took place during three years among the students of the Department of Informatics after the completion of programming final test at the end of the semester. In the last phase we realized national research among the selected group of secondary and university teachers of Informatics.

Figures in next chapter present findings from research phase (we obtain 38 responses), where students firstly attempt traditional examination (programming in paper, oral examination, without computer evaluation) and next semester attempt on-line testing according to described model. Mostly we use Likert type statements with five choices ranged from “absolutely agree (1)” to “absolutely disagree (5)”. Some overall presented outcomes are also from other phases, 94 responses from students with only traditional experiences and 97 responses from students with only computer examination experiences.

Results of Research

Visibility of the passing/remaining time was considered to be a stressful factor, which make rise of the nervousness. Negative responses were directed to some of test settings, such as hiding of full results with feedback immediately after test completion (it only displays the achieved score). Preview of itemized feedback and assessment was possible only in the presence of the teacher.

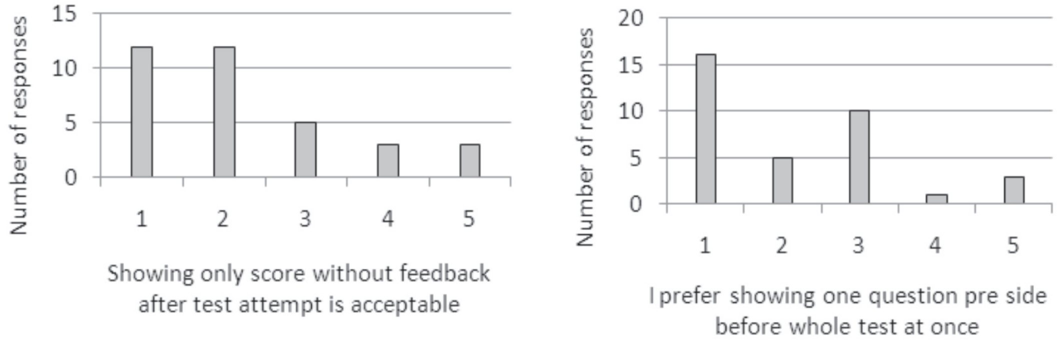


Figure 4. Responses on Likert statements, choices ranged from "absolutely agree (1)" to "absolutely disagree (5)".

The responses indicate that the most favourable seems to be a formal and user site of Moodle system. Also Palmárová (2009) present, that students found using CMS Moodle to be the essential complement to the traditional face to face instruction, they appreciated the online course a lot.

Most views on the disadvantages were directed to automatic evaluation of the short answers without compromise. "The man may evaluate a little yes or a little no." Even more, any clerical error may be evaluated incorrectly.

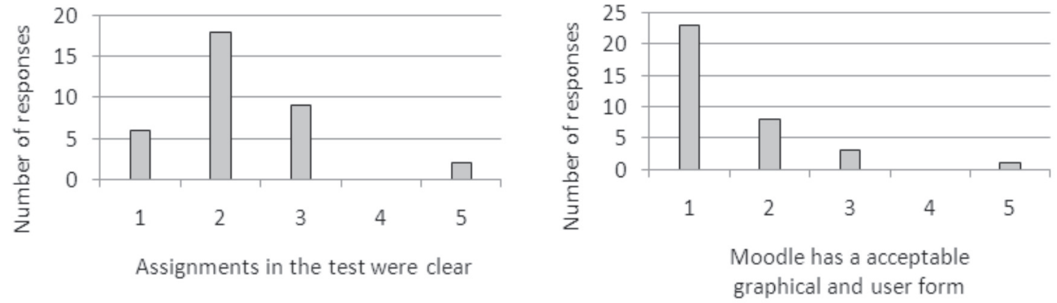


Figure 5. Responses on Likert statements, choices ranged from "absolutely agree (1)" to "absolutely disagree (5)".

The first year students usually do not have any experiences with electronic learning environment and electronic testing and most of them had no previous programming experience at all. Lack of both experiences might have acted a major inhibitor on their failure in final exam. Therefore we recommended to create a group of self-tests with the same conditions as the final test, which facilitate students to adapt their activity. During semester we make available two on-line tests to exercise. The first of them was static without answers presentation. The second was dynamic, generated randomly from little question database with automatic feedback (the same type like final test). Next figures present the students opinion to them, we used choices: (1) Yes a lot (2) More yes like no, (3) More no like yes, (4) No at all, (5) I did not make a look on it before my exam attempt.

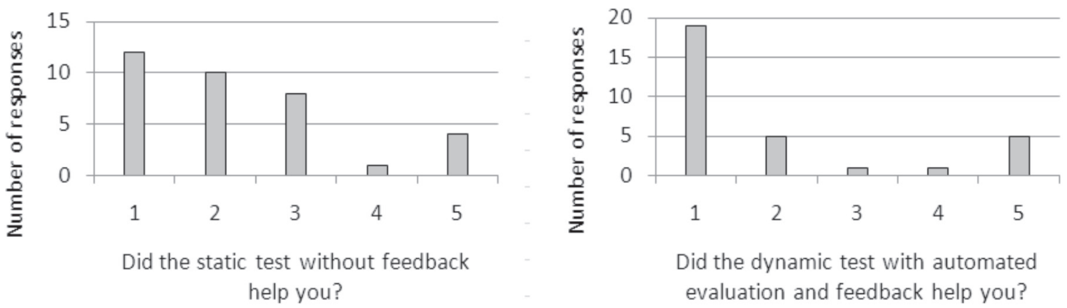


Figure 6. Responses on statements about exercise tests. Scale describe above.

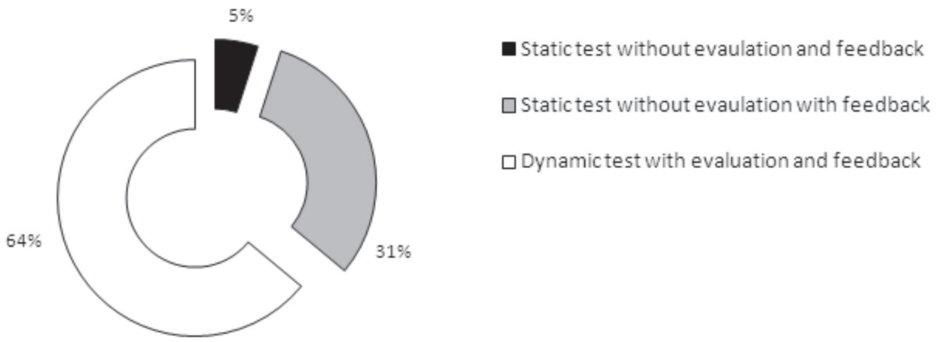


Figure 7. Responses on question: "Which kind of exercise test do you prefer for further usage?"

At the end we were interested whether the result of final test without previous self-test manipulating will be the same. About 43 % of students absolutely disagree with this statement, only 3 % absolutely agree and 31 % was in the middle.

According to the responses in the survey conducted among teachers of Informatics in middle and high schools, we could conclude that the elaboration of content into the categories, specification of their objectives and set up of answer score for questions are clear, correct and reasonable.

Some teachers propose to merge question categories, some terms were considered to be unreasonable for testing. On the other hand there were also very positive comments, which showed a balance in questions creating. Teachers finally taught the test results show students comprehensive knowledge in programming theory and partially practise. Recommendation of further usage of electronic tests is unquestionable.

From open answers (essay) is clear that a model is sufficient for classification only if it is accompanied by practical tests. It is truly supplement by mandatory, optional and bonus practical tasks that are rated by teachers continuously during the semester.

Conclusions

The automated testing should be treated as sensitive as the use of any other methods of teaching and testing. Positive perception of electronic testing is clear, about 70 % of student like it more than traditional way. But the same students prefer combine form of testing, where teacher still plays the essential role. Electronic testing may even indirectly discriminate disadvantaged students (e.g, blind or partially sighted). Therefore we should also think about the alternative option, such as distributing tests in paper form (printed according to electronic variant). It helps student to use a different compensation devices.

We analyzed the possibilities of systems suitable for testing and enabling simple archiving of results and their statistical processing and we finally chose LMS Moodle as the most accessible alternative for educational organizations.

Automating of testing may be perceived as an effort to replace tutors by machines. However today it is still not realizable, because we cannot ensure the objectivity during entire process. The survey shows that students and even teachers demand to involve teachers into the electronic testing process. We realize this due to possibility of personal consultations between teacher and student immediately after the test attempt.

Choosing the right type of test question has big importance in the computerized testing. The most contradictory are perceived tasks on which evaluation some batch of reasoning is required. Most of the students such type of questions would let check to teachers, despite of testing by computer.

None of the forms of testing will always be fully acceptable by the whole group of students so it is appropriate forms according to the occasion properly rotate.

Note

This is a revised and expanded version of an international scientific conference paper in *Problems of Education in the 21st Century*. The paper was recommended for republication by the conference scientific committee.

The reference for the Conference version is:

Capay, M. (2010). Electronic Automated Evaluated Tests in the Subjects of Programming. In. *DIVAI 2010 - Distance Learning in Applied Informatics* (Conference Proceedings, Nitra, Slovakia, May 4-6, 2010). Nitra: Constantine the Philosopher University in Nitra, p. 195-199.

References

Cápay, M. (2009a). Model of computer aided knowledge testing in the subject programming 1. *DIDMAT-TECH 2008. 1st part.* Eger: Esterházy Károly College.

Cápay, M. (2009b). *Development and use of intelligent computer systems to support the teaching of informatics subjects* (Doctoral dissertation, University of Constantine the Philosopher). Nitra.

Cápay, M. (2010a). Electronic Automated Evaluated Tests in the Subject of Programming. *DIVAI 2010 - Distance Learning in Applied Informatics*. Nitra, pp. 196-199.

Cápay, M. (2010b). Model of computer aided knowledge testing in the subject programming 2. *DIDMAT-TECH 2009.* Trnava: Trnavská Univerzita.

Kipper H., & Rüütmann, T. (2010). Strategies and techniques of questioning effectuating thinking and deep understanding in teaching engineering at Estonian centre for engineering pedagogy. *Problems of Education in the 21st Century*, Vol. 19, pp.36 - 44.

Kučera, P., & Kvasnička, r. & Vydrová, H. (2008). Evaluation of test questions using the item analysis for the credit test of the subject of mathematical methods in economics in the Moodle LMS. *Efficiency and Responsibility in Education*. Praha: Czech University of Life Sciences in Prague, p. 145 - 153.

Palmárová, V. (2008). Best practices for using CMS Moodle in introductory programming courses. *DIDMATTECH 2008. 1st part.* Eger: Esterházy Karóly College.

Palmárová, V., & Lovászová G. (2007) Elektronické portfólio a jeho využitie vo vyučovacom procese. *Technológia vzdelávania,* 3, p. 13-15.

Glusac, D., & Radosa, D. & Karuovic, D., & Dragicaivin. D. (2007) Pedagogical and Didactic-Methodical Aspects of E-learning. 6th WSEAS International Conference on E-ACTIVITIES, Spain, 2007, pp 67-75.

Adviced by Martin Bilek, Hradec Kralove University, Czech Republic

Martin Cápay Assistant professor, Constantine the Philosopher University, Faculty of Natural Sciences, Department of Informatics , Tr. A. Hlinku 1, 949 74 Nitra, Slovakia.
E-mail: mcapay@ukf.sk
Website: http://www.ki.fpv.ukf.sk/en/personal-pages/martin-capay