A. CHUKHRAY, E. YASHINA, O. LESHCHENKO

National Aerospace University "Kharkiv Aviation Institute", Ukraine

FORMALIZATION OF TASKS GENERATION FOR COMPLEX OF INTERACTIVE WEB-TESTS ON MATH

*The approach to the formalization of the generation of problem situations applicable to the development of tutoring programs consisting of many tasks is considered. The main errors arising during the software-based generation of parameters are specified. Mathematical modeling of parametrical generation algorithms examined by examples of tasks that make up complex tests on mathematics for secondary schools. The parametric generation method proposed in the article allows getting the large quantitative variations in task problem situations. Thereby, every learner will get a personal unique set of tasks. The structure and functionality of web-tests complex consisting of tasks generated via the proposed method are described. The **subject** of research in the article is the process of computer training in mathematics. The **goal** is to develop a method for task generation for mathematical disciplines. **Tasks**. Research and analysis of the set of mathematical problems. Parameterization of each task and development method and algorithms for automated generation parameters with the determination of incorrect combinations of parameters or problem situations that have no solution. Estimation of borders of admissible for approximate answers. Evaluation of the user solution of a single task and a sequence of tasks. The general objective of the work is to make the software product consisting of a sequence of mathematics tasks. The software should have an extended user interface for the graphical presentation of various problem situations in various mathematical topics. The program must be accessible via the Internet. The following **results** were obtained: developed methods and algorithms of task generation, which provide correct problem situations and unique parameter sets for each user; described the program complex structure and developed the software system of mathematical web-tests provides two levels of difficulty. **Conclusion**. The scientific novelty lies in the development of the method of task generation for interactive web tests on the mathematics and its computer implementation with the possibility of graphical representation of tasks and checking of tasks correctness.*

Keywords: computer tutoring programs; tasks generation algorithms; mathematical modelling.

Introduction

Computer tutoring programs on mathematics (CTPM) as a tool of knowledge and abilities transferring from teacher to learner are in increasing demand in the modern world. At the same time with the growth of interest in them, requirements for CTMPs became more severe.

During the evolution process, they have been turning from simple testing software and electronic textbooks to modern intelligent tutoring environments [1, 2]. This system can individualize learning by individual student parameters adjustment [3]. Now a lot of software tutoring systems are developed. Software systems for mathematical tasks presentation use relationship schemas [4], flowcharts schemas, elements of game behavior [6, 7].

Tutoring system developing by using artificial intelligence, system dynamics [8], *probabilistic approach [9, 10], and other methods. A lot of studies are dedicated to the effectiveness of* intelligent tutoring environments and their benefits for teachers and students [11, 12].

The direction connected with the automatic generation of tasks completed by pupils is perspective science objective [13, 14]. During work on this, there is an offer of some difficulties like requirement of task generation with capability to check right pupil competence components and acceptance requirement of tasks in terms of pedagogues. The last one is frequently connected with restrictions of calculations without the calculator, uniqueness of the answer, integer answers reception, etc.

This paper describes the experience of authors in mathematical modeling of tasks generation or CTMP. Models are received during work on two projects: the website of interactive tests on the mathematician for preparation to the external independent estimation of graduates in comprehensive schools and complex interactive web-tests on the mathematician. For the last one, the description is given.

Problem statement

Pre-analysis of tasks that are necessary for computer realization has revealed the most widespread sit-

uations which can lead to wrong system functionality. Functional errors are an illegal generation of parameters in the task, answer presence already in a problem situation (direct or indirect), the wrong estimation of correct answer inputted in nonspecific for system form, and many situations when right answer finding is an impossible task. Let's call problem situations with such errors of generation "incorrect" and without errors – "correct".

Parameterization of each task is a specific feature of work. Also, generated parameters must be without such values which can provoke irrational value of answer or value with a large number of signs after the decimal separator. It is connected with the inhibition of machine calculations during the completion of some tasks by the user.

In some tasks input of the assessed answer is supposed. In other words, the answer should be entered approximately. Thus there was a problem to provide in these tasks borders of admissible input for the right answer. The problem of a choice of a maximum deviation at answer input should be solved for each task.

The general objective of work is to make a program product consisting of a sequence of tasks that can be accessed through Internet. It is necessary to anticipate all adverse exceptions which can arise at a design stage of software for each task.

In addition, one of the important problems is to provide such generation of parameters values, when it is convenient to show these values in visual interactive form. Also, these values must be visually recognizable by the user accurately or can be estimated qualitatively or quantitatively (following task requirements).

To each task from the sample, the individual approach including the construction of a mathematical model of the task according to its features has been applied.

Finally, we can specify the following subtasks, which determine features of mathematical models construction process for web-tests complex:

– task problem situations should be generated by such method providing maximal similarity of computer tasks to their real prototypes presented in «Key Stage 3» papers [1];

– parameters should be generated automatically, so each user receives the unique set of test tasks;

– generation of parameters values should lead to forming only of correct tasks and task problem situations;

– during the generation of parameters values it is necessary to consider whether calculator usage is allowed for the user;

– mathematical formulation of tasks generation must consider the possibility of their computer implementation;

– mathematical formulation of tasks generation must consider such factors, as the possibility of answer correctness check and estimation of it;

– in many tasks with wide GUI, mathematical modeling must consider the capability of graphical representation of values in problem situation of task.

Formalisation of tasks generation

For constructivism of the further statement we will consider a technique of generation formalisation by the concrete example.

Let's consider a task with problem situation: «In a survey, X people were asked "What kind of newspaper did you buy today?". Results are: morning newspaper was bought by x_1 people, evening newspaper was bought by x_2 people and x_3 people bought no newspaper. Complete the pie chart to show this information. Pie chart already has one drawn sector in initial state [1].» Let's enter three additional values – a_1, a_2 and a_3. This values are equal to inner angles of pie chart corresponding to each value of x_1, x_1 and x_3. The sum of sectors inner angles values must be equal to 360 degree because whole pie chart is a circle, so:

$$\sum_{i=1}^{n} a_i = 360(\deg). \qquad (1)$$

The number of people in each respondent's category sum must be equal to total number of respondents:

$$\sum_{i=1}^{n} x_i = X. \qquad (2)$$

Also, following relations between this values must be kept:

$$\frac{a_1}{x_1} = \frac{a_2}{x_2} = ... = \frac{a_n}{x_n} = \frac{360^o}{X} = k. \qquad (3)$$

Value **n** in equations (1), (2) and (3) is a number of categories in the survey. Values of parameters a_i must be not null positive integers for providing convenient user's plotting of diagram sectors on application of computer mouse. Values of parameters **x**, must be not null positive integers too, as numbers of respondents.

Plotting of only one sector on the pie chart is sufficient for checking of learner's skills of diagram structuring. Plotting of more than one sector leads checking the same knowledge and abilities. So, we can use restriction for problem situation with n=3. Let's get mathematician formulation for this situation. Solution of the task can be simplified when some angles of sectors are equal. So, to

avoid such situation we accept inequality $a_1 < a_2 < a_3$ as true. Than we must select value of a_1, less then 120deg and value of a_2 less than 180deg. Let's designate rounding operation as [...], then:

$$\forall a_1 \in \{1,...,119\}, \forall a_2 \in \{a_1+1,...,179\};$$

$$a_3 = 360 - a_1 - a_2;$$

$$\forall a_2 < a_3,$$

$$\forall x_1 \in \{1,...,\alpha_1 - 1\};$$

$$k = \frac{a_1}{x_1},$$

$$x_2 = \frac{a_2}{k},$$

$$x_3 = \frac{a_3}{k}.$$

if $|[x_2]- x_2| < 10\text{-}3$, then $x_2 = [x_2]$;

if $|[x_3]- x_3| < 10\text{-}3$, then $x_3 = [x_3]$;

if $x_2 = [x_2] \wedge x_3 = [x_3]$, then

$S = (a_1, a_2, a_3, x_1, x_2, x_3)$ – tuple of parameters satisfied all conditions.

As a result of described method of generation, we get all possible variants of parameters values when problem situation is correct. Such variants should be 5573 for integer k and 11069 for any k.

Thereby parameterization provide sufficient increment of task variants number. One of generation variants when n=3, k=2, x1=71, $x_2=15$, $x_3=94$, $a_1=142$ deg, $a_2=30$ deg, $x_3=188$ deg and X =180 are

showed on fig. 1. This variant was derived after program generation based on described method.

Let's consider another task example. Problem situation in a general form can be formulated:

«The diagram shows a triangle $\triangle XYZ$ (see fig. 2). Side XY is of length $k_{aXY} \cdot a + k_{bXY} \cdot b$, side XZ is of length $k_{aXZ} \cdot a + k_{bXZ}$ and side YZ is of length a. The triangle $\triangle XYZ$ is isosceles with XY = XZ. The perimeter of the triangle is P. Find the values of a and b». Mathematical formulation of task can be represented as:

$$\begin{cases} k_{aXY} \cdot a + k_{bXY} \cdot b = k_{aXZ} \cdot a + k_{bXZ} \cdot b, \\ k_{aXY} \cdot a + k_{bXY} \cdot b = k_{aXZ} \cdot a + k_{bXZ} \cdot b + a = P. \end{cases} \quad (4)$$

We can get following inequality from first equation in (4) system:

$$\frac{a}{b} = \frac{k_{bXZ} - k_{bXY}}{k_{aXY} - k_{aXZ}} \quad (5)$$

Let's select a and b values from set of non-negative integers. Coefficients in second member of equation we select from integers. Let's select values of a and 6 from [2;20] interval and values of k_{bXY} and k_{aXZ} from [-10;10] interval. Also it is necessary to take into account restrictions $k_{bXZ} \neq k_{bXY}$ and $k_{aXY} \neq k_{aXZ}$ required for solvable of task (first equality in (4) must be not identity). If we suppose that numerators and denominators in first and second members
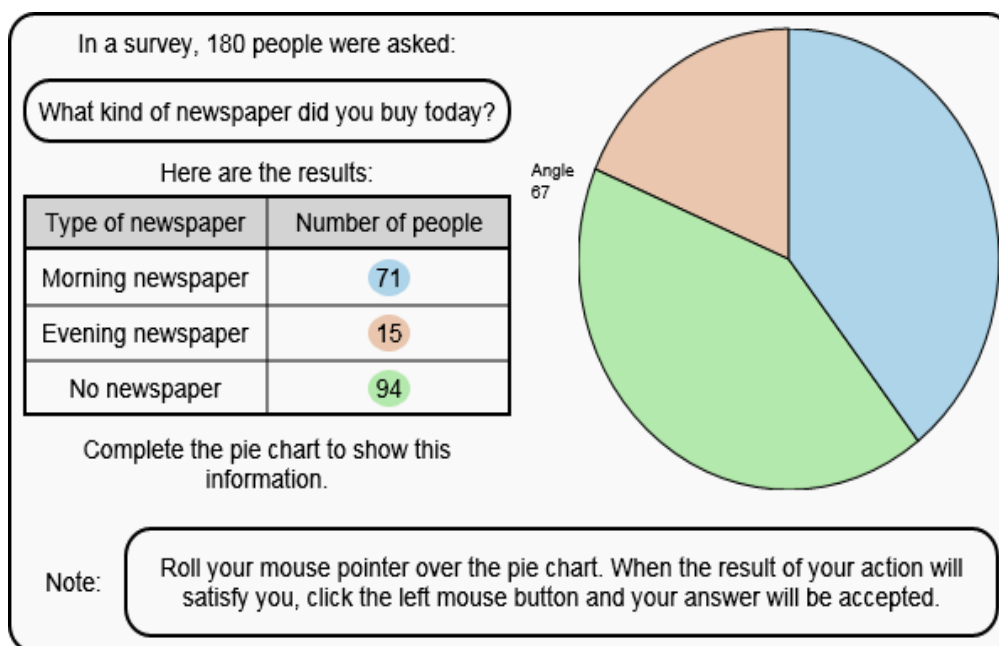

Fig. 1. Test task on plotting of pie chart sectors

The diagram shows a triangle

Side XY is of length **+6b**
Side XZ is of length **9a-30b**
Side YZ is of length **a**

The triangle is **isosceled** , with **XY = XZ**
The perimeter of the triangle is **80**

Find the values of a and b

a = [____]
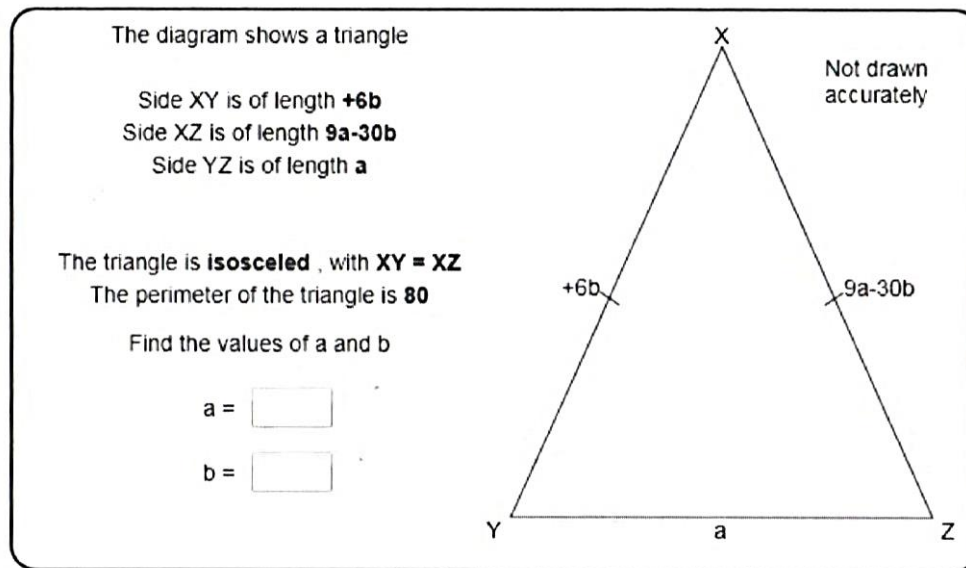
b = [____]

X
Not drawn
accurately

+6b

9a-30b

Y a Z

Fig. 2. Test task on sides lengths of isosceles triangle calculation

of equation (5) are equal, generally equations for calculation of a and b can be offered as:

$$a = k_{bXZ} - k_{bXY},$$
$$b = k_{aXY} - k_{aXZ}. \qquad (6)$$

No matter how strict it is, number of possible generation variants will be sufficiently great and generation of a same parameters for two students in classroom probability will be small (this will be proved lower). Value of P can be calculated on second equation in system (4), as all required values in this equation are known. Using mathematical formulation, we can write:

$$\forall \alpha \in \{2,...,20\}, \forall k_{bXY} \in \{-10,...,10\}, k_{bXY},$$
$$k_{bXZ} = a + k_{bXY};$$
$$\forall b \in \{2,...,20\}, \forall k_{aXZ} \in \{-10,...,10\}, k_{aXZ},$$
$$k_{bXY} = b + k_{bXZ},$$
$$P = k_{aXY} \cdot a + k_{bXY} \cdot b + k_{aXZ} \cdot a + k_{bXZ} \cdot b + a.$$

Computer simulation of such generation method produces N = 144399 different variants of problem situations. Let us assume that complex of tasks is used by 100 learners in one classroom. Then it is possible to calculate probability of situation when the same problem situation of task is generated at least on two computers. This probability can be found as:

$$P = \frac{(100-1)}{N} = \frac{99}{144399} \approx 0.0006856. \quad (7)$$

So, it is probability of 0.06 % for situation when at least two learners get tasks which are generated identically. In number of tasks, generation isn't complicated

by severe restrictions and relations between parameters. Method of all answer components generation (calculation) along with partial method was used for these tasks. For example, for task on fig. 3, in which it is required to write missed numerals in numbers completing right equality, only two summands are generated. Main restriction during generation is grounded on fact that result of sum must be three-digit number.

Such methods were used for generation of more than 40 different tasks.

Description of web-tests complex

The user must install an internet browser supporting Silverlight movies on his computer for the usage of web-complex. He must open the internet browser and input the address [6] to start working. On the loaded page, the user can select the difficulty level of tasks.

After level choice, the web browser loads page consisting of a sequence of tasks in Silverlight movies form. At the bottom part of this page, there is a functional button for confirmation of the user's answers' correctness. When a user pushes this button, a page with information about the results of the tasks passage must be loaded. After viewing of results user can back to the main page and try to pass test tasks again.

The structure of complex shown in fig. 4. On the figure, the main pages are in complex structure marked as paper pages with turned-in angles. Under pages, blocks are rectangular blocks representing elements of the user interface placed at corresponding pages. Rectangular blocks with vertical lines on the left side presents program server modules responsible for generation, storing, and providing data corresponding to clients' requests. Dashed relationship lines show possible crossing between pages.

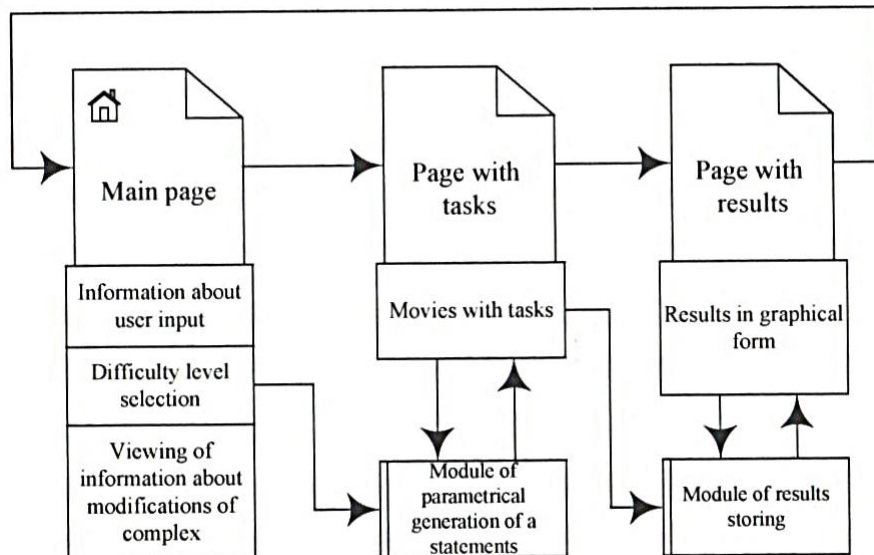Fig. 3. Test task on structuring of right equality

Fig. 4. Complex structure

Solid relationship lines show the direction of data exchange between structural elements of the complex. The server side of the complex consists of program modules and the client-side is an internet browser with one of the complex pages opened.

Developed program complex of mathematical web-tests consists of two main pages with movies containing tasks (fig. 5) and dynamic page with a graphical representation of task passage results.

The system provides two levels of difficulty (4th and 8th levels). For each of them, its own set of knowledge is formed.

Tasks are implemented as separate multimedia Silverlight movies and are located descending on the page. In many tasks besides text input from the keyboard, there is a possibility to interactive input by mouse.

During this, the element's image is changing in response to user actions. This approach simplifies the understanding of some mathematical laws, presents them in a good understanding visual form. For example, in job scaling and drawing figures, these actions can be performed by dragging the point of the manipulator and watching the results on the screen.

It was developed movies with tasks on working with diagrams, on figures scaling, on algebra, on statistics, on working with a rectangular coordinate system, on angles measurements, on the structuring of sequences, on geometry, on probability theory, and another.

Mode, when it is necessary to get a selection of identical tasks for a group of learners, is provided. For getting such selection user must input into the request string an additional parameter named "testvar" with a unique value for each selection. For this sample, it was decided not to develop separate algorithms for generating each of the tasks, and expanding already existing for the generation of jobs is arbitrary.

After clicking on the "End test" button internet browser will be linked to the page with the results of the tasks passage. The table with three columns showed on this page. The first column contains cells with numbers of tasks, the second one contains letters of tasks (usually, the task consists of a few sub-tasks marked by letters a, b, c, and so on). The third column includes cells with images indicating the correctness of the answer for the associated task/subtask. If the answer is correct, there is a green image in the cell and if the answer is wrong, there is a white cross inside the red circle image in the cell. In the left bottom corner of the page, the statistic box is situated. Statistic box is a fixed positioning panel with information about learner mark, percentage of right answers and link to the main page. Learner mark is calculated as a sum of marks for each task in the set, which
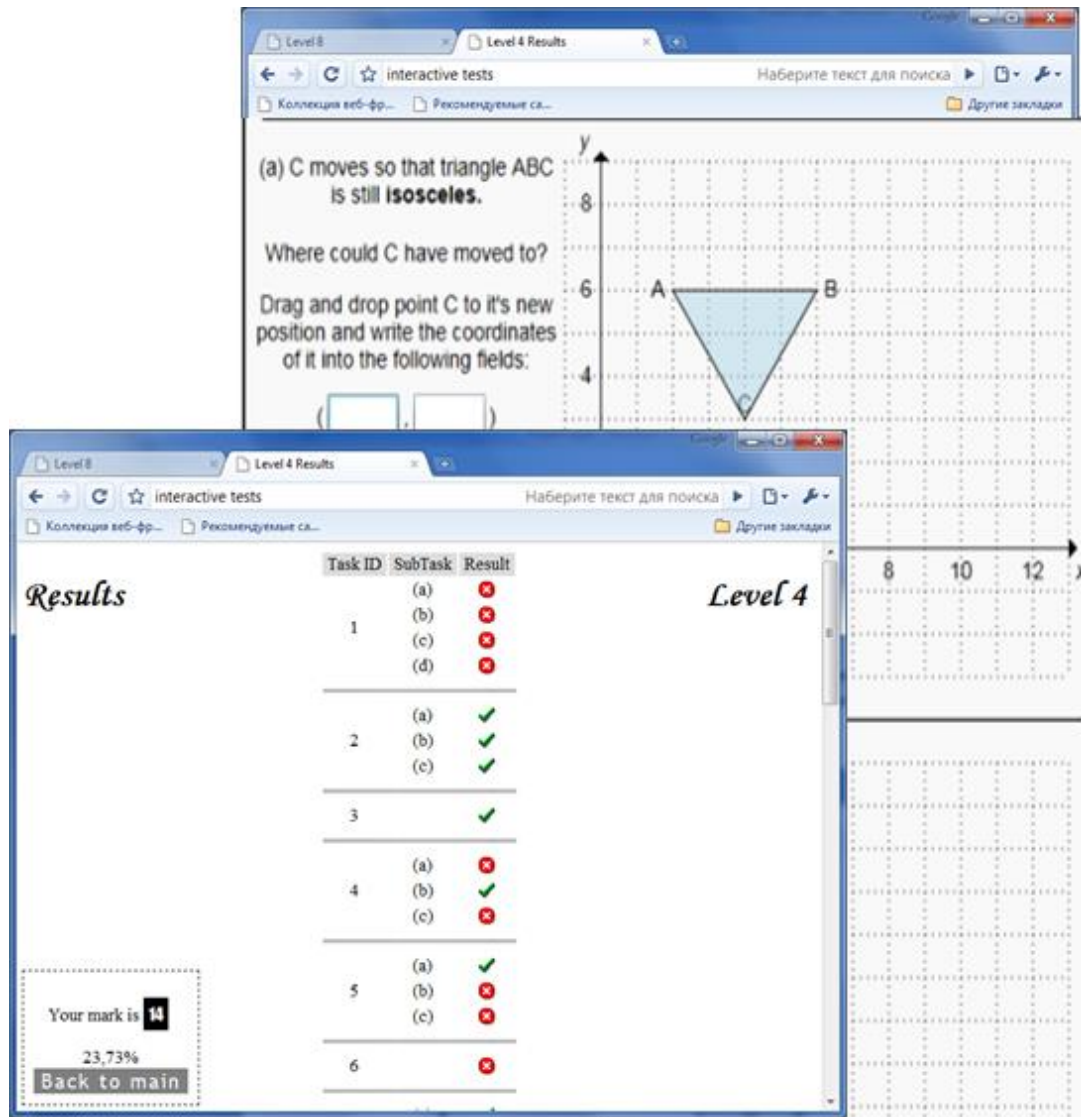
Fig. 5. General view of web-tests complex

was proposed to the user on page with tasks. For each task, the solution learner may get from one to five points. So, it is the max number of points available for the user inset. The percentage of correctness is calculated as the ratio of user points to total points available for the set of tasks.

Conclusions and perspectives

During the analysis and mathematical modeling of tasks selected for complex mathematical web-tests, have been separated general parameters, which are included in tasks. For the value of each parameter was determined special interval of generated values. This allows developing such methods and algorithms of generation, which provides correct problem situations. During this development restrictions formulated in the problem statement section of this paper must be satisfied. These

restrictions are the possibility of computer implementation, the possibility of checking of tasks' correctness, and the possibility of graphical representation of tasks.

In the future, it is planned to integrate the math web-tests system with other software tutoring products, for example, the intelligent tutoring system for SQL [15]. It is also planned to expand the functionality by the implementation of dynamic bayesian nets models, which are successfully used for engineering skills training [16].

For testing and errors made during modeling identification, prototypes of each task were implemented in interactive Silverlight movies form. After debugging these prototypes joined in a single complex meant for learners testing (http://zno-kharkiv.org.ua/cimt/). This complex is a part of the project completed for CIMT (Centre for Innovation in Mathematics Teaching) at the University of Plymouth, Great Britain.

References (GOST 7.1:2006)

1. Intelligent Tutoring System Trends 2006-2018: A Literature Review [Text] / J. Han, W. Zhao, Q. Jiang, M. Oubibi, X. Hu // 2019 Eighth International Conference on Educational Innovation through Technology (EITT). – IEEE, 2019. – P. 153-159. DOI: 10.1109/EITT.2019.00037.

2. Open learner models in supporting self-regulated learning in higher education: A systematic literature review [Text] / D. Hooshyar et al. // Computers & Education. – 2020. – Vol. 154. – P. 103-108. – Article Id: 103878. DOI: 10.1016/j.compedu.2020.103878.

3. Melis, E. ActiveMath: An intelligent tutoring system for mathematics [Text] / E. Melis, J. Siekmann // International Conference on Artificial Intelligence and Soft Computing. – Springer, Berlin, Heidelberg, 2004. – P. 91-101. DOI: 10.1007/978-3-540-24844-6_12.

4. Teaching Underachieving Algebra Students to Construct Models Using a Simple Intelligent Tutoring System [Text] / K. VanLehn, F. Milner, C. Banerjee, J. Wetzel // International Conference on Artificial Intelligence in Education. – Springer, Cham, 2021. – P. 367-371. DOI: 10.1007/978-3-030-78270-2_65.

5. A flowchart-based intelligent tutoring system for improving problem-solving skills of novice programmers [Text] / D. Hooshyar, R. B. Ahmad, M. Yousefi, F. D. Yusop, S.-J. Horng // Journal of Computer Assisted Learning. – 2015. – Vol. 31, Iss. 4. – P. 345-361. DOI: 10.1111/jcal.12099.

6. A solution-based intelligent tutoring system integrated with an online game-based formative assessment: development and evaluation [Text] / D. Hooshyar, R. B. Ahmad, M. Yousefi, M. Fathi, A. Abdollahi, S. J. Horng, H. Lim // Educational Technology Research and Development. – 2016. – Vol. 64, no. 4. – P. 787-808. Eric Id: EJ1108006.

7. How does order of gameplay impact learning and enjoyment in a digital learning game? [Text] / Y. Wang, H. Nguyen, E. Harpstead, J. Stamper, B. M. McLaren // International Conference on Artificial Intelligence in Education. – Springer, Cham, 2019. – Vol. 11625. – P. 518-531. DOI: 10.1007/978-3-030-23204-7_43.

8. Using system dynamics to model student performance in an intelligent tutoring system [Text] / M. T. Sanz, D. Arnau, J. A. González-Calero, M. Arevalillo-Herráez // Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization. – 2017. – P. 385-386. DOI: 10.1145/3079628.3079635.

9. Corbeil, J. P. Probabilistic Approaches to Detect Blocking States in Intelligent Tutoring System [Text] / J. P. Corbeil, M. Gagnon, P. R. Richard // Proceedings on 16th International Conference on Intelligent Tutoring Systems, Athens, Greece, June 8–12, 2020. – Vol. 12149. – P. 79-88. DOI: 10.1007/978-3-030-49663-0_11.

10. Chukhray, A. Proximate Objects Probabilistic Searching Method [Text] / A. Chukhray, O. Havrylenko // Integrated Computer Technologies in Mechanical Engineering. Advances in Intelligent Systems and Computing. – 2020. – Vol. 1113. – P. 219-227. DOI: 10.1007/978-3-030-37618-5_20.

11. San Pedro, M. O. Z. Carelessness and affect in an intelligent tutoring system for mathematics [Text] / M. O. Z. San Pedro, R. S. J. d. Baker, M. M. T. Rodrigo // International Journal of Artificial Intelligence in Education. – 2014. – Vol. 24, Iss. 2. – P. 189-210. DOI: 10.1007/s40593-014-0015-y.

12. Feng, M. Predicting State Test Scores Better with Intelligent Tutoring Systems: Developing Metrics to Measure Assistance Required [Text] / M. Feng, N. T. Heffernan, K. R. Koedinger // Intelligent Tutoring Systems. ITS 2006. Lecture Notes in Computer Science. – Springer, Berlin, Heidelberg, 2006. – Vol. 4053. – P. 31-40. DOI: 10.1007/11774303_4.

13. Чухрай, А. Методология обучения алгоритмам [Текст] : моногр. / А. Чухрай. – Харьков : Нац. аэрокосм ун-т им. Н. Е. Жуковского "Харьков. авиац. ин-т", 2017. – 336 с.

14. Structural diagnosis method for computer programs developed by trainees [Text] / D. Gaydachuk, O. Havrylenko, J. P. Martínez Bastida, A. Chukhray // Proceedings of 15 th International Conference on ICT in Education, Research, and Industrial Applications. – Kherson, Ukraine, 2019. – P. 485-490.

15. Chukhray, A. The method of student's query analysis while intelligent computer tutoring in SQL [Text] / A. Chukhray, O. Havrylenko // Радіоелектронні і комп'ютерні системи. – 2021. – № 2. – С. 78-86. DOI: 10.32620/reks.2021.2.07.

16. Chukhray, A. The engineering skills training process modeling using dynamic bayesian nets [Text] / O. Havrylenko, A. Chukhray // Радіоелектронні і комп'ютерні системи. – 2021. – № 2. – С. 87-96. DOI: 10.32620/reks.2021.2.08.

References (BSI)

1. Han, J., Zhao, W., Jiang, Q., Oubibi, M. and Hu, X. Intelligent Tutoring System Trends 2006-2018: A Literature Review. *2019 Eighth International Conference on Educational Innovation through Technology (EITT)*, 2019, pp. 153-159, DOI: 10.1109/EITT.2019.00037.

2. Hooshyar, D., Pedaste, M., Saks, K., Leijen, Ä., Bardone, E., Wang, M. Open learner models in supporting self-regulated learning in higher education: A systematic literature review. *Computers & Education*, 2020, vol. 154, pp. 103-108, Article Id: 103878. DOI: 10.1016/j.compedu.2020.103878.

3. Melis, E., Siekmann, J. ActiveMath: An Intelligent Tutoring System for Mathematics. *International Conference on Artificial Intelligence and Soft Computing - ICAISC 2004. Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2004, vol. 3070, pp. 91-101. DOI: 10.1007/978-3-540-24844-6_12.

4. VanLehn, K., Milner, F., Banerjee, C., Wetzel, J. Teaching Underachieving Algebra Students to Con-

struct Models Using a Simple Intelligent Tutoring System. *International Conference on Artificial Intelligence in Education. AIED 2021. Lecture Notes in Computer Science*, Springer, Cham, 2021, vol. 12749, pp. 367-371. DOI: 10.1007/978-3-030-78270-2_65.

5. Hooshyar, D., Ahmad, R. B., Yousefi, M., Yusop, F. D. and Horng, S. J. A flowchart-based intelligent tutoring system for improving problem-solving skills of novice programmers. *Journal of Computer Assisted Learning*, 2015, vol. 31, iss. 4, pp. 345-361. DOI: 10.1111/jcal.12099.

6. Hooshyar, D, Ahmad, R. B., Yousefi, M, Fathi, M, Abdollahi, A., Horng, S. J., Lim, H. A solution-based intelligent tutoring system integrated with an online game-based formative assessment: development and evaluation. *Educational Technology Research and Development,* 2016, vol. 64, no. 4, pp. 787-808. Eric Id: EJ1108006.

7. Wang, Y., Nguyen, H., Harpstead, E., Stamper, J., McLaren, B. M. How Does Order of Gameplay Impact Learning and Enjoyment in a Digital Learning Game? *International Conference on Artificial Intelligence in Education. AIED 2019. Lecture Notes in Computer Science*, Springer, Cham, 2019, vol. 11625, pp. 518-531. DOI: 10.1007/978-3-030-23204-7_43.

8. Sanz, María T., Arnau, David., González-Calero, José A., Arevalillo-Herráez, Miguel. Using System Dynamics to Model Student Performance in an Intelligent Tutoring System. *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (UMAP '17). Association for Computing Machinery, New York, NY, USA, 2017, pp. 385-386. DOI: 10.1145/3079628.3079635.

9. Corbeil, J. P., Gagnon, M., & Richard, P. R. Probabilistic Approaches to Detect Blocking States in Intelligent Tutoring System. *International Conference on Intelligent Tutoring Systems*, Springer, Cham, 2020, vol. 12149, pp. 79-88. DOI: 10.1007/978-3-030-49663-0_11.

10. Chukhray, A., Havrylenko, O. Proximate objects probabilistic searching method. *Integrated Computer Technologies in Mechanical Engineering. Advances in Intelligent Systems and Computing.* Springer, Cham, 2020, vol. 1113, pp. 219-227. DOI: 10.1007/978-3-030-37618-5_20.

11. San Pedro, M. O. Z., Baker, R. S. J. d., Rodrigo, M. M. T. Carelessness and Affect in an Intelligent Tutoring System for Mathematics. *International Journal of Artificial Intelligence in Education*, 2014, vol. 24, iss. 2, pp. 189-210. DOI: 10.1007/s40593-014-0015-y.

12. Feng, M., Heffernan, N. T., Koedinger, K. R. Predicting State Test Scores Better with Intelligent Tutoring Systems: Developing Metrics to Measure Assistance Required. *Intelligent Tutoring Systems*. ITS 2006. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2006, vol. 4053, pp. 31-40. DOI: 10.1007/11774303_4.

13. Chukhray, A. *Metodologiya obucheniya algoritmam : monografiya* [Methodology for learning algorithms, monograph]. Kharkiv, National Aerospace University "KhAI" Publ., 2017. 336 p.

14. Gaydachuk, D., Havrylenko, O., Martínez Bastida, J. P., Chukhray, A. Structural diagnosis method for computer programs developed by trainees. *Proceedings of 15 th International Conference on ICT in Education, Research, and Industrial Applications*, Kherson, Ukraine, 2019, pp. 485-490.

15. Chukhray, A., Havrylenko, O. The method of student's query analysis while intelligent computer tutoring in SQL. *Radioelektronni i komp'uterni sistemi – Radioelectronic and computer systems*, 2021, no. 2, pp. 78-86. DOI: 10.32620/reks.2021.2.07.

16. Chukhray, A., Havrylenko, O. The engineering skills training process modeling using dynamic bayesian nets. *Radioelektronni i komp'uterni sistemi – Radioelectronic and computer systems*, 2021, no. 2, pp. 87-96. DOI: 10.32620/reks.2021.2.08.

ФОРМАЛІЗАЦІЯ ГЕНЕРАЦІЇ ЗАВДАНЬ ДЛЯ КОМПЛЕКСУ ІНТЕРАКТИВНИХ ВЕБ-ТЕСТІВ З МАТЕМАТИКИ

А. Г. Чухрай, О. С. Яшина, О. Б. Лещенко

Розглянуто один із підходів до формалізації генерації проблемних ситуацій, застосовних для розробки навчальних програм, що складається з низки завдань. Вказано основні помилки, що виникають під час генерації параметрів на основі програмного забезпечення. Математичне моделювання алгоритмів параметричного покоління досліджується на прикладах завдань, що складають комплекс тестів з математики для загальноосвітніх шкіл. Запропонований у статті метод параметричного генерування дозволяє отримати великі кількісні варіації ситуацій проблемної задачі. Таким чином кожен учень отримає індивідуальний унікальний набір завдань. Описано структуру та функціональність комплексного веб-тестування, що складається із завдань, створених запропонованим методом. Предметом дослідження у статті є процес навчання комп'ютера з математики. Мета-розробити метод формування завдань з математичних дисциплін. Завдання. Дослідження та аналіз сукупності математичних задач. Параметризація кожного завдання та розробка методу та алгоритмів для параметрів автоматизованої генерації з визначенням неправильних комбінацій параметрів або проблемної ситуації, які не мають рішення. Оцінка меж допустимого для приблизних відповідей. Оцінка вирішення користувачем окремого завдання та послідовності завдань. Загальною метою роботи є створення програмного продукту, що складається з послідовності математичних завдань. Програмне забезпечення повинно мати розширений користувальницький інтерфейс для графічного представлення різноманітних про-

блемних ситуацій з різних математичних тем. Програма повинна бути доступною через Інтернет. Були отримані такі результати: розроблено методи та алгоритми генерації завдань, які забезпечують правильні проблемні ситуації та унікальний набір параметрів для кожного користувача; описав складну структуру програми та розробив програмну систему математичних веб-тестів, що забезпечує два рівні складності. Висновок. Наукова новизна полягає у розробці методу формування завдань для інтерактивних веб-тестів з математики та його комп'ютерної реалізації з можливістю графічного представлення завдань та перевірки правильності виконання завдань.

Ключові слова: комп'ютерні програми навчання; алгоритми формування завдань; математичне моделювання.

ФОРМАЛИЗАЦИЯ ГЕНЕРАЦИИ ЗАДАЧ
ДЛЯ КОМПЛЕКСА ИНТЕРАКТИВНЫХ ВЕБ-ТЕСТОВ ПО МАТЕМАТИКЕ

А. Г. Чухрай, О.С. Яшина, А. Б. Лещенко

Рассмотрен один из подходов к формализации генерации проблемных ситуаций, применимый при разработке обучающих программ, состоящих из ряда заданий. Указаны основные ошибки, возникающие при программной генерации параметров. Математическое моделирование алгоритмов параметрической генерации исследуется на примерах задач, составляющих комплекс тестов по математике для общеобразовательных школ. Предлагаемый в статье метод параметрической генерации позволяет получать большие количественные вариации постановок задач. Таким образом, каждый ученик получит индивидуальный уникальный набор заданий. Описаны структура и функционал комплекса веб-тестов, состоящего из задач, генерируемых предложенным методом. Предметом исследования в статье является процесс компьютерного обучения математике. Цель - разработать методику генерации заданий по математическим дисциплинам. Задания. Исследование и анализ множества математических задач. Параметризация каждой задачи и разработка методов и алгоритмов автоматической генерации параметров с определением неверных комбинаций параметров или проблемных ситуаций, не имеющих решения. Оценка границ допустимых для примерных ответов. Оценка решения пользователем отдельной задачи и последовательности задач. Общая цель работы - создать программный продукт, состоящий из последовательности математических задач. Программное обеспечение должно иметь расширенный пользовательский интерфейс для графического представления множества проблемных ситуаций по различным математическим темам. Программа должна быть доступна через Интернет. Были получены следующие результаты: разработаны методы и алгоритмы генерации задач, обеспечивающие корректные проблемные ситуации и уникальный набор параметров для каждого пользователя; описана структура программного комплекса и разработана программная система математических веб-тестов, обеспечивающая два уровня сложности. Заключение. Научная новизна заключается в разработке метода генерации заданий для интерактивных веб-тестов по математике и его компьютерной реализации с возможностью графического представления заданий и проверки правильности заданий.

Ключевые слова: компьютерные обучающие программы; алгоритмы генерации задач; математическое моделирование.

Чухрай Андрій Григорович – д-р техн. наук, доцент, зав. каф. математичного моделювання та штучного інтелекту, Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут», Харків, Україна.

Яшина Олена Сергіївна – канд. техн. наук, доц., доц. каф. комп'ютерних наук та інформаційних технологій, Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут», Харків, Україна.

Лещенко Олександр Борисович – канд. техн. наук, проф., доц. каф. комп'ютерних наук та інформаційних технологій, Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут», Харків, Україна.

Andrey Chukhray – Doctor of Technical Sciences, Associate Professor, Head of the Department mathematical modelling and artificial intelligence, National Aerospace University «Kharkiv Aviation Institute», Kharkiv, Ukraine,
e-mail: achukhray@gmail.com, ORCID: 0000-0002-8075-3664.

Elena Yashina – Candidate of Technical Science, Associate Professor of the Department of Computer Science and Information Technology, National Aerospace University «Kharkiv Aviation Institute», Kharkiv, Ukraine,
e-mail: o.yashina@khai.edu, ORCID: 0000-0003-2459-1151.

Oleksandr Leshchenko – Candidate of Technical Science, Professor, Associate Professor of the Department of Computer Science and Information Technology, National Aerospace University «Kharkiv Aviation Institute», Kharkiv, Ukraine,
e-mail: o.leshchenko@khai.edu, ORCID: 0000-0001-9405-4904.