

Three-subject didactic model for teaching algorithmization and programming online

Michael Lvov¹, Hennadiy Kravtsov¹, Ludmila Shishko¹ and Olha Hniedkova¹

¹*Kherson State University, 27 Universytetska Str., Kherson, 73003, Ukraine*

Abstract

Distance education plays an important role in the context of the COVID-19 pandemic and the growing geopolitical challenges. Electronic educational resources (EER), in particular distance learning courses, are widely used in both secondary and higher education. Requirements for the quality of EER become relevant, especially for ensuring of students' independent activity at practical tasks. For students of the "Software Engineering" specialty, courses in algorithmization and programming are basic. The feature of the "Programming" discipline is the presence of a significant number of tasks on programming mathematical tasks. An analysis of educational literature showed there are practically no thematic tasks books that have important characteristics: a task books is devoted to mathematical tasks programming and is an elementary introduction to computer algebra; supports the most common programming languages in sports programming Pascal, C/C++, Python, etc. The paper proposes the model of learning pedagogical system in the three-subject didactics paradigm. On the example of the course "Practice of Programming Olympiad Tasks" the system of registration and monitoring of mathematical tasks programming using wiki technology is considered. Within the framework of the proposed model of learning pedagogical system, the subsystems of the use of didactic materials of lectures, a task book, practical exercises and laboratory works are described. The methods and tools of organizing support for students' independent activity are illustrated using the example of a workshop for solving Olympiad tasks in programming.

Keywords

Distance learning, online course, programming of mathematical tasks

1. Introduction

The problems of creating and effective use of distance learning courses, especially in negative situations when it is impossible to conduct classroom classes, play an important role. The set of these problems is the subject of theoretical research and practical work outlined in this research. The feature is the continuation of the research outlined in our work [1] and the further use of didactic materials (DM) that are highlighted there, in the developing and conducting of online basic **courses in algorithmization and programming**, using the course "**Olympiad Programming Tasks Practice**" for students of the Faculty of Computer Sciences, Mathematics and Physics of Kherson State University.

The subject of the study is the distance learning system that supports the developing, use and maintenance of online electronic educational resources (EER) of the distance course "Programming mathematical problems. First steps" [2]. The feature of the distance course (DC) is the combination of studying theoretical algorithms with practical programming of mathematical tasks, the library of which consists of author's tasks and tasks of participants in the educational process. Automation of creation and storage of tasks in the EER library is one of the tasks of this research. The developed model of the learning system consists of a subsystem of didactic lecture materials use, support for practical classes and laboratory work, and support for independent work of students.

A review of the scientific-methodical and educational literature shows that there are many textbooks and problem collections in algorithmization and programming for high school and junior students. First of all, let's note the books that are included in the golden fund of the world's scientific and educational literature in algorithmization. This is a five-volume encyclopedia of algorithms by Knuth [3, 4], in which


CTE 2023: 11th Workshop on Cloud Technologies in Education, December 22, 2023, Kryvyi Rih, Ukraine

✉ lvovms14@gmail.com (M. Lvov); kgmkherson@gmail.com (H. Kravtsov); lshishko59@gmail.com (L. Shishko); gnedkova84@gmail.com (O. Hniedkova)

ORCID 0000-0002-0876-9928 (M. Lvov); 0000-0003-3680-2286 (H. Kravtsov); 0000-0001-9346-4338 (L. Shishko);

0000-0001-5194-2194 (O. Hniedkova)

© 2024 Copyright for this paper by its authors.

 Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

we especially note the first two volumes, an educational monograph by Dijkstra [5], an interesting book by Bentley [6], a unique book by Graham et al. [7] devoted to the mathematical foundations of computer science, and also books by Wirth [8] and other classics. Among the recently published ones, we note the book by Cormen et al. [9].

According to the content, EER is an elementary introduction to computer mathematics. Theoretical aspects of such branch of science as computer algebra are systematically described in [10, 11, 12]. There are many books on the market in the most popular systems of computer algebra, such as Mathematica, Maple, Mathcad [13, 14, 15].

In Ukraine, as in other countries, various programming competitions are held for schoolchildren and students, which are called sports programming. That is why books by authors who are specialists in sports programming and activists of the Ukrainian Olympic movement deserve special attention. The materials of the All-Ukrainian Programming Olympiads were included in many problem books, for example, [16, 17].

2. Description of the pedagogical system and problem statement

The practical results of this study were obtained at teaching of disciplines in algorithmization of programming developed at the Department of Computer Sciences and Software Engineering of Kherson State University using the following educational software systems:

- Software-methodological complex “Video Interpreter of Search and Sorting Algorithms” [18].
- Collection “Programming of mathematical tasks. First steps” [1].
- Software complex “Web OAP Fundamentals of Algorithmization and Programming” in the distance learning system (DLS) “KSU online” [19],

and at studying the following disciplines:

- Programming. Programming language C.
- Programming. programming language Python.
- Fundamentals of algorithmization and programming.
- Algorithms and data structures.
- Workshop on solving Olympiad problems.
- Computer algebra fundamentals.

Didactic materials of DC “Programming of mathematical tasks. First steps” have the following programmatic structure:

- The distance course is a complex of interrelated software training modules Textbook, Task book, mathematical tasks editor.
- The textbook consists of thematic sections of mathematical tasks and methodical instructions for solving mathematical tasks in the format of algorithms in the form of program code.
- The task book for each thematic section contains a number of tasks with their solutions: the text of the task, a description of the solution algorithm, and program code in three programming languages: Pascal, C/C++, and Python.
- The math problem editor provides users with the ability to create new tasks in the Task book. Distance course support is based on maintaining the quality of tasks developed in the EER library thanks to wiki technology use [20].

The purpose of the research is to study the teaching methodology and software of the distance course “Programming of mathematical tasks. First steps”, namely to investigate

1. Description of the pedagogical system and construction of its model based on the interaction of subjects of the educational process.
2. Subsystems description of the didactic materials use of the Task book.
3. Description of the registration subsystem and the use of new tasks in the collection.

3. Description of the pedagogical system and problem statement

Our approach is based on the paradigm of the three-subject didactics [21], which consists in an effective combination of methods of lecture courses (direct teacher-student feedback), conducting practical classes and laboratory work (current feedback), control (final control), independent activity (self-control) of the student with didactic materials and other sources of educational information.

The learning process model is presented in figure 1.

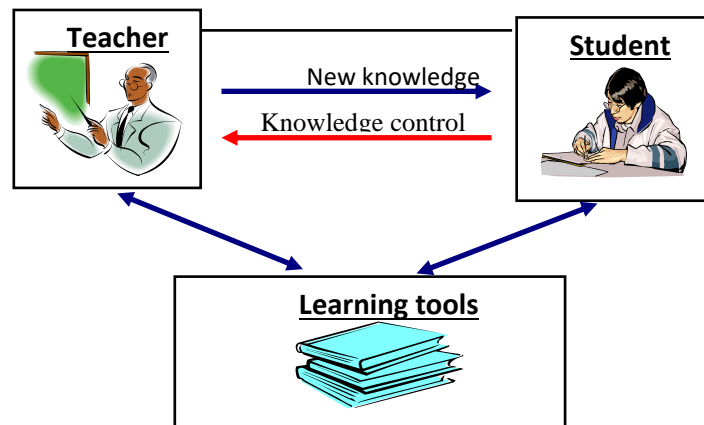


Figure 1: The learning process model.

The learning process is the main educational process; all other educational processes are subordinate to its implementation. The learning process is complex in terms of content and a long-term process, for its implementation the significant human, material and financial resources are involved, and the state infrastructure has been created. Therefore, there is every reason to consider the learning process as a business process.

The main tasks of the program system of educational purpose:

1. Relevance ensuring, availability and quality of learning tools.
2. Supporting of the process of new knowledge transferring.
3. Supporting of the process of students' independent activity.
4. Knowledge control support (feedback).

Our approach is based, on the one hand, collections of problems in algorithmization and programming are basic, supporting didactic materials for studying several topics, and, on the other hand, each topic relies on several problems in the collection. The example of the key thesis.

Example 1 shows the ratio *Task (algorithm)* \leftrightarrow *Topic of the discipline*.

Example 1. Ratio Task (Algorithm) \leftarrow Topic of discipline

Topic: Floyd's Algorithm

- Shortest path problem.
- Method of dynamic programming.
- Oriented graph adjacency matrix.
- Estimation of algorithm complexity by time according to the control structure.
- Python programming language: attached management operators.

Ratio Topic of discipline \leftarrow Task (Algorithm)

Topic: Divide and Posses programming method

- Search for a fake coin using simple scales.
- Binary search in an ordered list.

- Algorithm for sorting lists by merging.
- Karatsuba's algorithm for multiplying polynomials.

So, at developing a course for learning a certain discipline from the basics of algorithmization and programming, programming problems and material that explains their solution are those didactic units that make up the actual basis of one or another didactic material.

From our point of view, the online course in the basics of algorithmization and programming contains the following types of educational work:

- lectures,
- practical training,
- laboratory work,
- control,
- independent activity.

So, the main methodological and technological requirements and basic architectural solutions to the software system for educational purpose, which supports the formation and use of programming tasks as reference units of didactic materials, are considered.

4. Subsystem of didactic materials using of the Task book

Effective use of electronic Task books, first of all, requires the creation of a subsystem that supports the didactic materials (DM) expansion of the Task book with new reference units. Let's consider the requirements for subsystems supporting the learning process.

- Requirements for the composition of the task as a reference unit of DM.
- Requirements for the process of the task quality maintaining as a reference unit of DM.
- Requirements for the process of supporting practical classes and laboratory works.
- Requirements for the process of supporting students' independent work.
- Requirements for the process of creating, editing and monitoring tasks in the EER library.

4.1. Subsystem of using didactic lecture materials

The peculiarity of lecture materials for educational disciplines in algorithmization and programming on arbitrary topics is that they are based on several algorithms. The lecture should contain the algorithms and their explanations. The presentation should also contain these algorithms. Finally, the package of didactic lecture materials should be supplemented with a video lecture. Thus, our educational software system should support the selection process of individual algorithms and programs and the compilation in a single file of the basis for the text of the lecture and presentation.

The package of lecture didactic materials (LDM) consists of the following didactic units:

- abstract of the lecture,
- full text of the lecture,
- presentation of lectures,
- video lectures.

From our point of view, the didactic provision of the lecture (DPL) of a certain discipline has the following structure:

The author (compiler) of the discipline

The title of discipline

The list is the discipline content

Name of the lecture**Lecture plan****Abstract of the lecture*****Link* Full text of the lecture****Examples*****Link* Presentation of the lecture*****Link* Video lectures**

Of course, we guide the lecturers – users of the system to create author’s courses. Therefore, the DPL support subsystem has the following functionality:

Functions of the author

- Function of registration Author – course (new course).
- Function of package title editing (List-content (Lecture title, lecture plan, lecture abstract)).
- Function of lecture completing with tasks from the collection.
- Functions of lecture text editing (filling with other materials).
- Functions of editing of lecture presentation (filling with other materials).
- Function of adding a video lecture to the LDM (must be performed by the author after the lecture, which is automatically recorded in a software system such as Zoom)
- Function of presentation of video lecture.

Functions of the student

- View of package title and select a lecture.
- Reading of the full lecture text.
- Using the presentation.
- Watch videos.

The functionality we have provided is actually standard. The exception is the *function of completing the lecture with tasks from the collection*, which makes a connection between the collection of tasks (Task book) and didactic lecture materials.

Let’s consider the structure of the Task book. The Task book contains a set of tasks structured by content. An example is the structure of the Task book “Programming of mathematical tasks. First steps”, full information about it [1] and concisely – the item “Overview of software systems ...”

The Task book is written in Ukrainian, English and Russian. It supports Pascal, C, Python programming languages. The first three sections are the main ones, the 4th and 5th are intended for independent work and control. The structure of the task from the main section is as follows:

Task No. Task condition.

Instructions for the solution

Program text

Lecturer can choose any element of the structure independently of others. Consider an example of a structured lecture.

Example 2. Lecture “Real-time algorithms”**Lecture 11. Real-time algorithms****Plan.**

1. Introduction. Concept of real-time algorithm.
2. Task 3.7. Search by pattern. Lexical analysis.
 - 2.1. The concept of a finite automaton.
 - 2.2. Construction of a finite automaton for the pattern search problem.

- 3.1. Solving problems of lexical analysis in real time
- 3.2. Task 3.4. Information system of the bank.
4. Syntactic analysis.
 - 4.1. Task 3.2. Correctly spaced parentheses.
 - 4.2. Task 3.3. Correctly constructed arithmetic expressions
5. Conclusion
6. Literature.

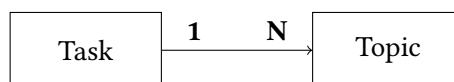
It should be noted that 95 percent of the full text of the lecture is made up of the tasks given in the plan. The conditions of the problems and instructions for solving them are given in full. Lecture presentations are formed semi-automatically. Slides, generated by the system contain only the conditions of the tasks. The lecturer has the opportunity to change the text of the lecture, forming other slides of the presentation.

Subsystem models of LDM support

The mathematical model of the LDM support process is a bipartite graph of ratio Task – Topic (Example 1)



The topic for the task is chosen by the user – the lecturer. Thus, the subgraph defining the :: “topicality” of ratio **Task** – **Topic** has the form



Therefore, the set of such connections is the result of the work of several lecturers, each uses this task for his own purpose.

Methodical assistance is provided by the sub-graph, which accumulates collective experience of the Task book using.



Therefore, the subsystem functionality should support editing (adding topics) and using both types of representation of bipartite graph **Task** – **Topic**.

The subsystem should also support the bipartite graph formation



At creating lecture didactic materials, if the lecturer includes author’s fragments in the text of the basic text of the lecture as a collection of tasks, the copyright protection function must be implemented.

Other functions to support the technology of selection, basis of the lecture, presentation and subsequent editing and use of the author’s lecture materials should be selected and implemented.

4.2. Support of subsystem for practical classes and laboratory works

The practical part of the educational process in the disciplines of algorithmization and programming can consist of such forms of classes as

- laboratory work,
- practical training,
- control,
- tests.

The main purpose of the practical part is to carry out reverse control of the assimilation of the theoretical provisions of the discipline. The distance form of education requires new approaches determined by the distance form itself.

From our point of view, regardless of the discipline, laboratory work is mandatory. Ideally, the set of programming tasks of the proposed structure is a sufficient source of didactic materials. However, the application of the tasks set in some disciplines requires a qualitative expansion of the topics. The example is a discipline “Algorithms and data structures”. Here are the mandatory topics:

Data structures and ATD algorithms *Lists, Queues, Stacks, Trees.*

Data structures and ATD algorithms *Insert, Delete, Find.*

Sort and search.

Algorithms on graphs.

Algorithms of computational geometry.

Ideally, the Task book should contain a sufficient number of tasks from these topics. We plan to expand the system of tasks in the direction of universalization.

Support of laboratory works. Note, the laboratory support subsystem must distribute, form individual tasks and check these tasks. However, these subsystem functions are standard and well-known, so we do not consider them.

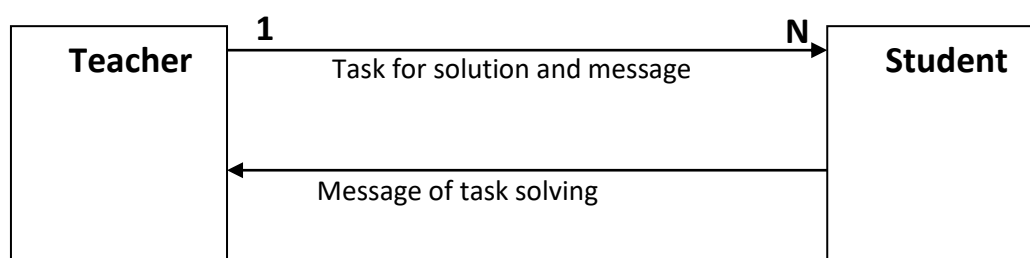
The basic sections for automatic distribution and formation of individual laboratory work packages are sections 4 and 5 of the Task book. The most difficult is the problem of checking the correctness of computer programs – the results of laboratory work. The standard approach is to create computer test packages for each task of these sections. This approach is implemented, in particular, in the Web OAP system [19]. However, this approach requires considerable time and quality control.

Support of practical classes. From our point of view, practical classes are an important form of the learning process in distance learning. Our practice has shown the importance of regular two-way communication between teacher and student. To implement this approach, it is suggested to plan a thematic plan of practical classes, 2 academic hours are assigned to each topic with the following functionality:

Topic of practical lesson

Instructional lesson. Teacher’s comments and instructions on general practical techniques for writing programs on this topic. Distribution of tasks for independent solution and preparation by each student of the subgroup of a short report on their task.

Final lesson. Introductory speech of the teacher. Brief messages from each of the students about the algorithm and program of their task. Discussions on the topic of the message. Practical training assessment.



For example, at studying disciplines:

- Workshop in Olympiad programming tasks solving
- Computer algebra fundamentals

The following tasks are considered on the specified topic.

Topic: Algorithms for solving elementary algebraic tasks

Tasks.

1. Solving of a system of linear inequalities of one variable with rational coefficients.
2. Search for rational roots of the reduced cubic equation.
3. Calculation of the integral coefficients of a cubic equation by its rational roots (Viet's theorem).
4. Analysis of a cubic equation with integer coefficients for the number of real roots
5. Analysis of a cubic equation with integer coefficients for the existence of roots of multiples of 2.
6. Solution of the reduced quadratic equation with integer coefficients in radicals
7. Solving a linear equation with modules
8. Karatsuba's algorithm of multiplying two polynomials with integer coefficients.

4.3. Independent activity support. Tests and control

Independent activity should be performed independently, without current control of its execution. The result of independent work is completed laboratory work and solved tasks of practical classes.

One of the most effective tasks of students' independent work is the task of a mathematical task creating and upload it to the EER library of the distance course "Programming of Mathematical Problem. First steps" [2].

The main stage of the learning process is the knowledge control process acquired by the student at the stages of explaining new material and practical work. Knowledge control is the form of feedback in the education system that ensures the quality of acquired knowledge as the main goal of the education system. The teacher, according to the discipline plan, has the opportunity to conduct tests and control. Ideally, feedback should be provided constantly, so monitoring of knowledge in various forms is part of the structure of any type of learning course. Without going into a detailed analysis of different types of control, it is important to control both declarative knowledge and procedural knowledge. Computer methods for controlling declarative knowledge are sufficiently developed [22]. In the IMS [23] and SCORM [24] standards, 20 types of test tasks are described, starting from the simplest test questions with an answer (YES, NO) and ending with complex questions. The answer is the educational task solving or laboratory work performing.

Tests should be used to control of theoretical knowledge. However, complex test tasks are open in the standard and are not supported by any of the known testing systems, since their implementation must be based on knowledge of a specific subject area and requires special efforts. Therefore, the construction of computer systems for testing productive knowledge, in particular, in the exact sciences, is an important problem. In [25], the approach based on the use of mathematical models and templates of test tasks is proposed. Unfortunately, this approach cannot be applied to the disciplines of algorithmization and programming. Therefore, the results of practical and laboratory work are the main form of control of students' success.

5. Subsystem of registration and use of new tasks in the Task book

The subsystem of registration and use of new tasks covers all three subjects of the learning process model. According to the paradigm of the three-subject didactics, the Learning tools element has the properties of a subject, because it has direct and reverse connections with other subjects of the system [21]. The Learning Tools are directly and feedback related to the Teacher and the Student (figure 2).



Figure 2: The subsystem of registration and use of new tasks in Task book.

As already mentioned above, the feature of DC is the presence of a significant number of mathematical tasks in the library of EER. The variability of these tasks can be high, because students have the opportunity to create new tasks or edit existing ones. In this way, students perform the function of quality control of the formulation of a mathematical task, the proposed algorithm for its solution, and the development of software code. The teacher has an expert role and he monitors the quality of tasks.

The participation of students in the developing and support of tasks in the EER library performs an important function of encouraging and increasing the motivation of students to intensify creative work in researching new algorithms for solving mathematical problems, improving the software code and ensuring the EER quality.

6. Conclusion

The pedagogical system is described and the learning process model of students of “Software Engineering” specialty in the disciplines of algorithmization and programming is developed. The paper proposes the model of the pedagogical learning system in the paradigm of the three-subject didactics. A feature of the discipline “Programming” is the presence of a significant number of tasks in programming of mathematical tasks. The problem with updating and creating new tasks is solved with the help of wiki technology. On the example of the course “Practice of Programming Olympiad Tasks” the system of registration and monitoring of programming of mathematical tasks is considered.

Within the framework of the proposed model of the pedagogical learning system, subsystems of the use of didactic materials of lectures, notebooks, practical and laboratory work are described. Using the example of a workshop on solving Olympiad programming tasks, the methodology and means of organizing support for students’ independent work are illustrated.

The research’s results are based on the experience of teaching disciplines in algorithmization and programming of mathematical tasks at the Department of Computer Sciences and Software Engineering of Kherson State University.

To assess the prospects for using the Web-application “Programming of Mathematical Tasks” in the educational process, the expert method was applied. The twelve experienced teachers in mathematics of Kherson secondary schools were interviewed. For evaluation, a five-point Likert system was chosen [1].

References

- [1] M. Lvov, H. Kravtsov, L. Shishko, O. Hniedkova, I. Chernenko, E. Kozlovsky, About Electronic Textbook “Mathematical Tasks Programming. First Steps”, in: O. Ignatenko, V. Kharchenko, V. Kobets, H. Kravtsov, Y. Tarasich, V. Ermolayev, D. Esteban, V. Yakovyna, A. Spivakovsky (Eds.), ICTERI 2021 Workshops, Springer International Publishing, Cham, 2022, pp. 334–349. doi:10.1007/978-3-031-14841-5_22.
- [2] M. Lvov, Mathematical Tasks Programming. Initial Steps, 2021. URL: <http://dls.ksu.ks.ua/dls/Library/LibdocView.aspx?id=dcb175b1-7ac0-4cb8-b97a-5448a2fca0f7>.
- [3] D. E. Knuth, The Art of Computer Programming, volume 1: Fundamental Algorithms, 3 ed., Addison-Wesley, 1997. URL: https://doc.lagout.org/science/0_Computer%20Science/2_Algorithms/

- The%20Art%20of%20Computer%20Programming%20(vol.%201_%20Fundamental%20Algorithms%20(3rd%20ed.))%20%5BKnuth%201997-07-17%5D.pdf.
- [4] D. E. Knuth, *The Art of Computer Programming, volume 2: Seminumerical Algorithms*, 3 ed., Addison-Wesley, 1997. URL: https://doc.lagout.org/science/0_Computer%20Science/2_Algorithms/The%20Art%20of%20Computer%20Programming%20%28vol.%202_%20Seminumerical%20Algorithms%29%20%283rd%20ed.%29%20%5BKnuth%201997-11-14%5D.pdf.
- [5] E. W. Dijkstra, *A Discipline of Programming*, volume 613924118, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1976.
- [6] J. Bentley, *Programming Pearls*, Addison-Wesley Professional, 1999.
- [7] R. L. Graham, D. E. Knuth, O. Patashnik, *Concrete mathematics: a foundation for computer science*, Addison-Wesley Professional, 1994. URL: https://doc.lagout.org/science/0_Computer%20Science/3_Theory/Mathematics/Concrete%20Mathematics.pdf.
- [8] N. Wirth, *Algorithms and Data Structures (Oberon version)*, 2004. URL: <https://web.archive.org/web/20110813043427/http://www.ethoberon.ethz.ch/WirthPub/AD.pdf>.
- [9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, *Introduction to Algorithms*, 4 ed., The MIT Press, Cambridge, Massachusetts and London, England, 2022. URL: <https://dl.ebooksworld.ir/books/Introduction.to.Algorithms.4th.Leiserson.Stein.Rivest.Cormen.MIT.Press.9780262046305.EBooksWorld.ir.pdf>.
- [10] A. G. Akritas, *Elements of Computer Algebra With Applications*, John Wiley & Sons, Inc., 1989.
- [11] P. Naudin, C. Quitté, *Algorithmique algébrique avec exercices corrigés*, Dunod, 1992.
- [12] J. H. Davenport, Y. Siret, É. Tournier, *Computer Algebra: Systems and Algorithms for Algebraic Computation*, 2 ed., Academic Press Ltd., 1988.
- [13] V. P. Dyakonov, *Mathematica 4.1 / 4.2 / 5.0 in mathematical and scientific-technical calculations*, SOLON-Press, Moscow, 2004.
- [14] V. P. Dyakonov, *Maple 9 in mathematics, physics and education*, SOLON-Press, Moscow, 2004.
- [15] V. Z. Aladjev, *Computer algebra systems: Maple: The art of programming*, 2006.
- [16] N. O. Herasymchuk, *Rozviazannia olimpiadnykh zadach z prohramuvannia (navchalnyi posibnyk dlia slukhachiv viddilennia kompiuternykh nauk MAN): navch. posib. dlia slukhachiv sektsii viddilennia kompiuternykh nauk*, VO MAN, Lutsk, 2010. URL: <http://vvman.lutsk.ua/file/k60.pdf>.
- [17] A. M. Hurzhii, V. V. Bondarenko, *Rekomendatsii do rozviazuvannia zadach Mizhnarodnykh i Vseukrainskykh olimpiad sered uchniv : navchalno-memodychnyi posibnyk*, Kyiv, 2008.
- [18] M. S. Lvov, O. V. Spivakovskiy, T. V. Zaitseva, H. M. Kravtsov, S. M. Kot, D. H. Kravtsov, K. S. Herasymenko, V. S. Peschanenko, A. Y. Hrabovskiy, A. O. Khoruzhenko, *Prohramno-memodychnyi kompleks "Videointerpretator alhorytmiv poshuku ta sortuvannia" ("Videointerpretator")*, 2002. URL: https://www.kspu.edu/About/DepartmentAndServices/DigitalInfrastructure/internationalprojects/NationalProjects/Videointerpreter_algorithms_for_search_n_sorting.aspx?lang=uk.
- [19] Kherson State University, *Services and Information Resources*, 2023. URL: <https://www.kspu.edu/Information/services.aspx?lang=en>.
- [20] E. O. Kozlovskiy, H. M. Kravtsov, *Resource of training "Wiki-document" in system "Kherson Virtual University"*, *Journal of Information Technologies in Education (ITE)* 15 (2013) 255–262. URL: <https://ite.kspu.edu/index.php/ite/article/view/309>.
- [21] A. Spivakovskiy, L. Petukhova, E. Spivakovska, V. Kotkova, H. Kravtsov, *Three-Subjective Didactic Model*, in: V. Ermolayev, H. C. Mayr, M. Nikitchenko, A. Spivakovskiy, G. Zholtkevych (Eds.), *Information and Communication Technologies in Education, Research, and Industrial Applications*, Springer International Publishing, Cham, 2013, pp. 252–273. doi:10.1007/978-3-319-03998-5_13.
- [22] M. S. Lvov, *Mathematical tests in computer math systems for educational purposes*, *Control Systems and Machines* (2011) 60–67.
- [23] 1EdTech Consortium, Inc, *Home | 1EdTech*, 2024. URL: <https://www.1edtech.org/>.
- [24] Rustici Software LLC, *SCORM.com HomePage: What is SCORM and How it Works*, 2024. URL: <https://scorm.com>.

- [25] M. Lvov, S. Kuzmenkov, H. Kravtsov, System for Testing Physics Knowledge, in: V. Ermolayev, F. Mallet, V. Yakovyna, H. C. Mayr, A. Spivakovsky (Eds.), *Information and Communication Technologies in Education, Research, and Industrial Applications*, Springer International Publishing, Cham, 2020, pp. 186–209. doi:10.1007/978-3-030-39459-2_9.