

КОМП'ЮТЕРНІ НАУКИ

УДК 640.412(477.83):005.591.6

DOI: [https://doi.org/10.32515/2664-262X.2023.7\(38\).1.3-9](https://doi.org/10.32515/2664-262X.2023.7(38).1.3-9)

Oksana Chernenko, Assos. Prof., PhD phys.&math. sci., **Nina Rudenko**, Senior Lecturer,
Daryna Bondar, applicant

Poltava University of Economics and Trade, Poltava, Ukraine

e-mail: oksanachernenko7@gmail.com, ninarudenko7@gmail.com, eldarychan@gmail.com

Development of simulator software on the topic "Normal algorithms" of the distance learning course "Theory of Algorithms"

The paper describes the design and development of a training simulator in the NetBeans integrated environment in the Java programming language. The simulator program articulates questions of three levels of complexity, methodological recommendations and theoretical issues on the topic. The developed software product is implemented in the corresponding distance learning course on the Moodle platform and is recommended for use in the educational process by applicants in the "Computer Science" specialty.

training simulator, educational activities, NetBeans IDE software development environment, Java programming language

Formulating the problem. With the development of technology, mankind has got many ways to make its life easier. Take, for example, the situations of the recent past when, due to the corona virus, it was not possible to work or study as usual. Then technologies came to the rescue that made it possible to do everything remotely - online meetings were used in training, and all kinds of tests or simulators were used to consolidate the material being studied.

Electronic simulators are generally indispensable in the training of some specialists, as they allow, in parallel with the study of theoretical material, to perform practical tasks, which increases the level of training. In addition, not always the applicant can perceive the information and understand the material being studied from the first presentation. This problem can be solved through the creation and implementation of various simulators in distance courses [1-6].

Analysis of recent research and publications. As part of the preparation of bachelor's and master's theses by students under the guidance of teachers, many simulators from various disciplines were developed and implemented in the educational process for students of the "Computer Science" specialty [7, 8].

Training simulators have proven themselves quite positively. Approaches to their design and implementation depend on the specifics of educational components and the amount of material being studied [9, 10]. Distance courses continue to be filled with new software. Due to the constant updating of the material from the educational component "Theory of Algorithms", there is a need to develop training simulators that allow users to learn theoretical material in parallel with testing knowledge and skills [11, 12, 13].

Setting objectives. Develop software for the simulator on the topic "Normal Algorithms", which contributes to improving the quality of education in higher educational institutions using distance technologies.

Presenting main material. The process of design and algorithmization preceded the software implementation of the simulator. After analyzing the theoretical material on this topic, it was decided to create a three-level training simulator with the possibility of accessing lectures before the actual testing. The training simulator is written in Java [14, 15] using the integrated environment and NetBeans.

When starting the program, a window opens to the user with the specified information about the developer, as well as three buttons that give the opportunity to go to theoretical information, choose the difficulty of testing or close the program.

If you click the "Information Page" button, a window with the theory will open, where you can familiarize yourself with the necessary information.

The "Start training" button opens a window with a choice of testing complexity.

The easy level includes 5 questions of a theoretical nature, the medium level includes 3 practical tasks, and the difficult level includes 2 practical ones.

Let's consider questions in the theoretical block of tasks.

Question 1. Words in Markov's theory are written...?

Answer options:

1. Small Latin letters p, q, r (or letters with indices).
2. Arbitrary characters.
3. Capital Latin letters P, Q, R (or letters with indices).

Question 2. Select matching function classes:

Respond options:

1. The class of all partially recursive functions.
2. The class of all computable functions.
3. The class of all recursive functions.
4. The class of all Turing-computable functions.
5. The class of all normally computable functions.
6. Class of elementary functions.

Question 3. The following words are given: P_1 : "vessel", P_2 : "ves", P_3 : "sel".

Choose the correct statement"

Answer options:

1. P_2 is a co-element of P_3 .
2. P_2 and P_3 are co-elements of P_1 .
3. P_1 is included in P_2 and P_3 .

Question 4. Establish a correspondence between statements and their meanings.

Statement: normal algorithm, Λ , word, alphabet.

Meaning: fixed set of characters; arbitrary non-empty set of characters; arbitrary set of characters; arbitrary sequence of letters; meaningful sequence of characters; an ordered final list of substitutions; finite list of substitutions; empty word.

Question 5. What does the Markov's normalization principle sound like?

Answer options:

1. There is an algorithm for finding the values of a function given in some alphabet if and only if the function is computable.
2. There is an algorithm for finding the values of a function given in some alphabet, when the function is normally computable.
3. There is an algorithm for finding the values of a function given in some alphabet if and only if the function is normally computable.
4. There is an algorithm for finding the values of a function specified in some alphabet when the function is computable.

If the user selects an incorrect answer, an error message and a hint are displayed.

Otherwise, if the correct answer is chosen, another window with the inscription "The answer is correct!" is displayed.

After the correct answer, the user can proceed to the next question.

After passing the last task of the specified level, a message appears about its successful completion, and the user returns to the start page, where the latter can select another level of testing.

Let's consider middle level tasks from the practical block.

Task 1. The normal algorithm in the alphabet $A = \{a,b,1\}$ is given by the scheme: $a \rightarrow 1, b \rightarrow 1$. Apply it to the words "11aab, baaabla".

At first, the panel contains only the condition and the first stage of the job. After the user enters the answer and confirms it with the button, an inscription about the result will appear below, as well as a field for the second part of the solution (Fig. 1)

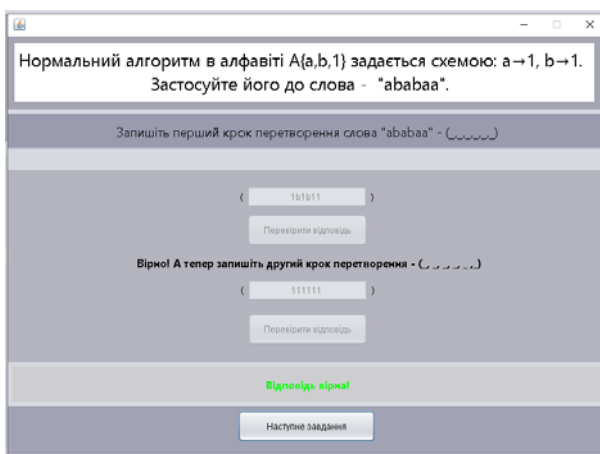


Figure 1 – Practical task 1

Source: compiled by the authors

Confirmation buttons block the corresponding fields. The next subtask with a different word becomes available only when both questions have been answered correctly.

Task 2. "The normal algorithm in the alphabet $A = \{a,b\}$ is given by the scheme: $ab \rightarrow a, b \rightarrow \Lambda, a \rightarrow b$. Apply it to the word – "abbbbaaab".

First, the user needs to go through the sequence of tasks from the top panel, choosing one answer among the four suggested (Fig. 2).

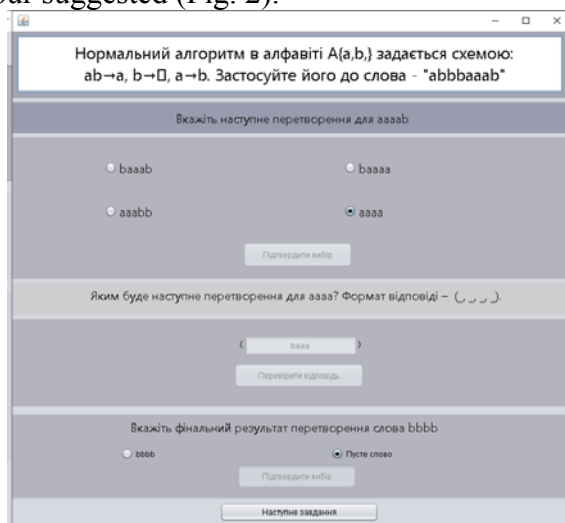


Figure 2 – Practical task 2

Source: compiled by the authors

After completing this stage, another one appears below, providing for the input of the answer from the keyboard. In case of an error, a hint for the correct answer will be displayed. If the choice is confirmed, the field is blocked, and then you can proceed to the sequence of questions of the last block in this task.

When the sequence is completed, the button to go to the last task of the middle level becomes available.

Task 3. “The normal algorithm is given by the scheme: $bb \rightarrow ba, ba \rightarrow a, a \rightarrow \Lambda, b \rightarrow \Lambda$.. Apply it to the word – *bbaab* .

At first, four buttons are available for selection, in the last two stages of the transformation, only two remain (Fig. 3).

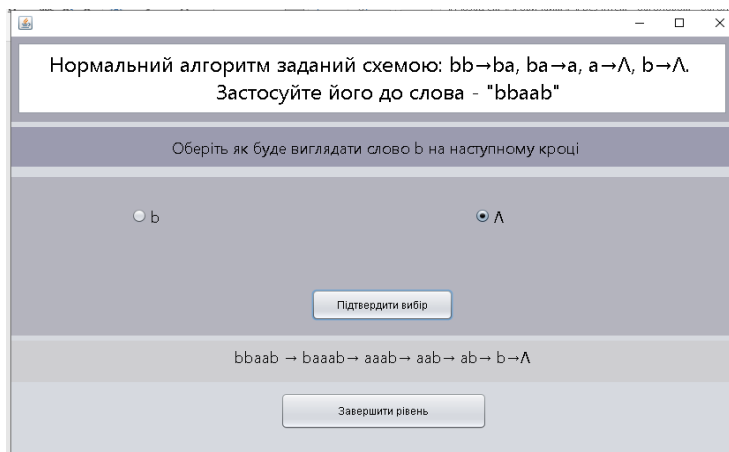


Figure 3 – Practical task 3

Source: compiled by the authors

When the selected option is correct, the transformation chain is built below, otherwise a hint box is displayed. The "Finish Level" button returns the user to the start page.

Let's move on to the consideration of problems of a complex level.

Task 1. "In the alphabet $B = A \cup \{a, b, c\}$, which is an extension of the alphabet A, consider the normal algorithm given by the scheme. Apply it to the following words: 999, 1998."

$0b \rightarrow 2$	$a0 \rightarrow 0a$	$0a \rightarrow 0b$	$0c \rightarrow 1$
$1b \rightarrow 3$	$a1 \rightarrow 1a$	$1a \rightarrow 1b$	$1c \rightarrow 2$
$2b \rightarrow 4$	$a2 \rightarrow 2a$	$2a \rightarrow 2b$	$2c \rightarrow 3$
$3b \rightarrow 5$	$a3 \rightarrow 3a$	$3a \rightarrow 3b$	$3c \rightarrow 4$
$4b \rightarrow 6$	$a4 \rightarrow 4a$	$4a \rightarrow 4b$	$4c \rightarrow 5$
$5b \rightarrow 7$	$a5 \rightarrow 5a$	$5a \rightarrow 5b$	$5c \rightarrow 6$
$6b \rightarrow 8$	$a6 \rightarrow 6a$	$6a \rightarrow 6b$	$6c \rightarrow 7$
$7b \rightarrow 9$	$a7 \rightarrow 7a$	$7a \rightarrow 7b$	$7c \rightarrow 8$
$8b \rightarrow c0$	$a8 \rightarrow 8a$	$8a \rightarrow 8b$	$8c \rightarrow 9$
$9b \rightarrow c1$	$a9 \rightarrow 9a$	$9a \rightarrow 9b$	$9c \rightarrow c0$
		$c \rightarrow 1$	
		$\Lambda \rightarrow a$	

When converting each word, a choice of one correct answer out of four is offered in stages. In case of an error, a hint will be displayed, allowing the user to answer correctly. If everything is correct, there is a transition to the next step and a word transformation chain is built below.

Task 2. Develop a normal algorithm that computes a function $f(w)=wu$ that adds a fixed word u to the right of each word w in the A alphabet. For the desired normal algorithm, the alphabet $B = A \cup \{2\} = \{0,1\} \cup \{2\} = \{0,1,2\}$ and a set of substitutions are given:

$21 \rightarrow 12$	$21 \rightarrow 20$
$\Lambda \rightarrow 1$	$2 \rightarrow .u$
$20 \rightarrow 2$	$20 \rightarrow 02$
$\Lambda \rightarrow u2$	$2 \rightarrow 2u$
$20 \rightarrow 01$	$\Lambda \rightarrow 2$

It is necessary to select the desired transformations from the proposed list and set them sequentially according to the specified pattern – ($_$, $_$, $_$, $_$). Using the result check the correctness of the constructed algorithm on specific examples.

If the correct transformations are selected from the list, then the transition to the next step takes place, that is, a panel appears with the second part of the task, where you need to transform the word according to the selected rules. Otherwise, an error message is displayed. Implementation of this task in the simulator is similar to the previous one.

As a result of successful completion of all tasks, a message is displayed to the user with the results of completion and a recommendation for better learning of the material.

Conclusions. Thus, a simulator on the topic "Normal Algorithms" of the discipline "Theory of Algorithms", compatible with the Moodle remote platform, was implemented in software. The training simulator has been introduced into the educational process of the Poltava University of Economics and Trade for students majoring in "Computer Science".

The software product is written in the NetBeans integrated environment in the Java programming language. In the future, it is planned to update the simulator program in accordance with changes in the curriculum of the discipline. The software product is used as a component of the information environment and provides an opportunity to approach the solution of the problem of effective integration of independent work into the educational process, allows filling the gaps in the knowledge of applicants for education.

Список літератури

1. Волкова, Н.П. Інтерактивні технології навчання у вищій школі: навчально-методичний посібник. Дніпро: Університет імені Альфреда Нобеля, 2018. 360 с.
2. Доценко, Н. Застосування навчальних комп'ютерних інтерактивних тренажерів здобувачами вищої освіти інженерних спеціальностей в умовах інформаційно-освітнього середовища. *Педагогічні науки: теорія, історія, інноваційні технології*. 2018. № 2(76). С. 118–128. URL: <https://doi.org/10.24139/2312-5993/2018.02/118-128> (дата звернення: 18.03.2023).
3. Virk R. The Simulation Hypothesis: An MIT Computer Scientist Shows Why AI, Quantum Physics and Eastern Mystics All Agree We Are In a Video Game. Bayview Books, 2019. 330 p.
4. Murray-Smith, D. J. Testing and Validation of Computer Simulation Models: Principles, Methods and Applications. Springer, 2015. 266 p.
5. Mohamed M Hafez and William E Tavernetti. Introduction To Computer Simulations For Integrated Stem College Education. WSPC, 2019. 234 p.
6. Vlachopoulos D., Makri A. The effect of games and simulations on higher education: a systematic literature review. *Intertional Journal of Education Technology in Higher Education*, 2017, Vol.14 (22). URL: <https://d-nb.info/1141475928/34> (Last accessed: 27.03.2023).
7. Черненко О.О., Чілікіна Т.В., Ольховська О.В. Розробка та використання навчальних тренажерів при підготовці фахівців напряму «Комп'ютерні науки». International scientific and practical conference "Mathematics, physics, mechanics, astronomy, computer science and cybernetics: issues of productive interaction": conference proceedings, Yuly 9-10. 2021. Wloclawek, Republic of Poland: "Baltija Publishing", 2021. С. 55-59.

8. Ольховська О.В., Собіборець О.Ю. Програмна реалізація елементів тренажеру з теми «Системи числення, арифметичні операції в різних системах числення» дисципліни «Архітектура обчислювальних систем». *Новітні інформаційно-комунікаційні технології в освіті: матеріали VII Всеукраїнської науково-практичної Інтернет-конференції молодих учених та студентів (Полтава, 24-25 листопада 2021 р.)*. Полтава: ПП «Астрая», 2021. С. 141-142.
9. Bhakat Rohit. Development of software simulator for the cut-off method of the distance learning course Elements of combinatorial optimization. *Комп'ютерні науки та інформаційні технології (КНІТ-2022): матеріали науково-практичного семінару*. 2022. Вип. 1. Полтава: кафедра КНІТ ПУЕТ, С.36-38.
10. Бардаченко С.Р. Електронний тренажер для дистанційного курсу «Теорія алгоритмів» на тему: «Машини Тьюрінга» URL : <http://www2.el.puet.edu.ua/st/mod/resource/view.php?id=98521> (дата звернення: 19.03.2023).
11. Яновський, А. Інформаційно-освітнє середовище в умовах дистанційного навчання. *Humanities science current issues*. 2020, Т. 4, № 30. С. 310-315. URL: <https://doi.org/10.24919/2308-4863.4/30.212627> (дата звернення: 19.03.2023).
12. Триус, Ю. В., Герасименко, І. В., Франчук, В. М. Система електронного навчання ВНЗ на базі MOODLE: метод. посіб. Черкаси, 2020. 220 с.
13. Ghrmida, S. M., Harkusha, S. V., Koshova, O. P., Orikhivska, O. G. Some Peculiarities of Development of Desktop Application «Using Array in Java». *Збірник наукових статей магістрів*. Полтава: ПУЕТ, 2022. С. 105-109.
14. Schildt H. Java: A Beginner's Guide, Eighth Edition. McGraw-Hill Education, 2018. 720 p.
15. Bloch J. Effective Java. Addison-Wesley Professional, 2018. 412 p.

References

1. Volkova, N.P. (2018). *Interaktyvni tekhnologii navchannia u vyshchii shkoli: navchalno-metodychnyi posibnyk [Interactive learning technologies in higher education: educational and methodological guide]*. Dnipro: Universytet imeni Alfreda Nobelia [in Ukrainian]
2. Dotsenko, N. (2018). Zastosuvannia navchalnykh kompiuternykh interaktyvnykh trenazheriv zdobuvachamy vyshchoi osvity inzhenernykh spetsialnostei v umovakh informatsiino-osvitnoho seredovyshcha [The use of educational computer interactive simulators by students of higher education in engineering specialties in the conditions of an informational and educational environment]. *Pedahohichni nauky: teoriia, istoriia, innovatsiini tekhnologii – Pedagogical sciences: theory, history, innovative technologies, 2(76)*, 118–128 [in Ukrainian].
3. Virk, R. (2019). The Simulation Hypothesis: An MIT Computer Scientist Shows Why AI, Quantum Physics and Eastern Mystics All Agree We Are in a Video Game. Bayview Books, 330 p. [in English].
4. Murray-Smith, D. J. (2015) Testing and Validation of Computer Simulation Models: Principles, Methods and Applications. Springer. 266 p. [in English].
5. Mohamed M Hafez & William E Tavernetti. (2019). Introduction To Computer Simulations for Integrated Stem College Education. WSPC. 234 p. [in English].
6. Vlachopoulos, D. & Makri, A. (2017). The effect of games and simulations on higher education: a systematic literature review. *Intertional Journal of Education Technology in Higher Education. Vol.14 (22)*. Retrieved from <https://d-nb.info/1141475928/34> [in English].
7. Chernenko, O.O., Chilikina, T.V. & Olkhovska, O.V. (2021). Rozrobka ta vykorystannia navchalnykh trenazheriv pry pidhotovtsi fakhivtsiv napriamu «Komp'uterni nauky» [Development and use of educational simulators in the training of specialists in the field of "Computer Sciences"]. Mathematics, physics, mechanics, astronomy, computer science and cybernetics: issues of productive interaction : *International scientific and practical conference (Yuly 9-10. 2021. Wloclawek): conference proceedings, Republic of Poland: Baltija Publishing. Pp. 55-59.* [in Ukrainian]
8. Olkhovska, O.V. & Sobiborets, O. Iu. (2021) Prohramna realizatsiia elementiv trenazheru z temy «Systemy chyslennia, aryfmetychni operatsii v riznykh systemakh chyslennia» dystsypliny «Arkhitektura obchysliuvalnykh system» [Software implementation of the elements of the simulator on the topic "Calculation systems, arithmetic operations in various calculation systems" of the discipline "Architecture of computing systems"]. The latest information and communication technologies in education: *VII Vseukrainska naukovo-praktychna Internet-konferentsia molodykh uchenykh ta studentiv (Poltava, 24-25 lystopada 2021 r.) – VII All-Ukrainian scientific and practical Internet conference of young scientists and students* (pp. 141-142). Poltava: PP «Astraia» [in Ukrainian].
9. Bhakat Rohit (2022). Development of software simulator for the cut-off method of the distance learning course Elements of combinatorial optimization. Computer science and information technologies (КНІТ-

- 2022): *Materialy Naukovo-Praktychnoho Seminaru – Materials of the Scientific and Practical Seminar* (pp. 36-38). Vol. 1, Poltava: kafedra KNIT PUET [in English].
10. Bardachenko, S.R. Elektronnyi trenazher dlia dystantsiinoho kursu «Teoriia alhorytmiv» na temu: «Mashyny Tiurinha» [Electronic simulator for the distance course "Theory of algorithms" on the topic: "Turing machines"] . *www2.el.puet.edu.ua*. Retrieved from <http://www2.el.puet.edu.ua/st/mod/resource/view.php?id=98521>.
 11. Yanovskyi, A. (2020). Informatiino-osvitnie seredovyshe v umovakh dystantsiinoho navchannia [Information and educational environment in the conditions of distance learning]. *Humanities science current issues. Vol. 4, 30, 310-315*. [in Ukrainian]
 12. Tryus, Yu. V., Herasymenko, I. V. & Franchuk, V. M. (2020) Systema elektronnoho navchannia VNZ na bazi MOODLE: Metodychnyi posibnyk [The MOODLE-based e-learning system of higher education institutions: Methodical manual]. Cherkasy. 220 p. [in Ukrainian]
 13. Ghrmida, S. M., Harkusha, S. V., Koshova, O. P., Orikhivska, O. G. (2022). Some Peculiarities of Development of Desktop Application «Using Array in Java». *Collection of scientific articles of masters*. Poltava: PUET [in English]
 14. Schildt, H. (2018). *Java: A Beginner's Guide, Eighth Edition*. McGraw-Hill Education. 720 p. [in English].
 15. Bloch, J. (2018). *Effective Java*. Addison-Wesley Professional. 412 p. [in English].

О.О. Черненко, доц., канд. фіз.-мат. наук, **Н.С. Руденко**, ст. викл., **Д.О. Бондар**, здобувач
Полтавський університет економіки і торгівлі, м. Полтава, Україна

Розробка програмного забезпечення тренажеру з теми «Нормальні алгоритми» дистанційного навчального курсу «Теорія алгоритмів»

Найважливішим завданням на сьогоднішньому етапі модернізації освіти в Україні є забезпечення якісного навчання під час очної, дистанційної чи змішаної форм для студентів навчальних закладів різного рівня акредитації. Для підвищення засвоєння теоретичного матеріалу та формування професійних компетентностей одним з ефективних методів є використання програм-тренажерів під час навчального процесу. У статті розглянуто процес програмної реалізації тренажеру дистанційного навчального курсу «Теорія алгоритмів».

У інтегрованому середовищі NetBeans на мові програмування Java створено програму-тренажер для дистанційного курсу «Теорія алгоритмів». Тренажер реалізовано у вигляді віконної програми, яка не потребує додаткового програмного забезпечення. Дослідження підтверджує, що саме інтерактивні додатки найбільш зрозумілі для користувачів та не потребують пояснень. Стартова сторінка програми містить інформацію про розробника; кнопки переходу до теоретичних відомостей; вибору рівня складності легкий, середній та складний; та завершення роботи. Легкий рівень включає у себе 5 запитань теоретичного характеру, у середньому - 3 практичних завдання, у складному - 2 практичних. Вгорі вікна програми знаходиться панель для відображення завдань, в центрі - варіанти або поля для запису відповідей, знизу - дві панелі, перша показує результати відповіді, а інша містить дві кнопки: «Перевірити відповідь» та «Наступне завдання». Задачі досить різноманітні: вибір однієї чи декількох правильних відповідей, завдання на відповідність та введення відповіді з клавіатури.

Розроблений програмний продукт виконує навчальну та контролюючу функції. Тренажер інтегровано в систему дистанційного навчання на платформі Moodle Полтавського університету економіки та торгівлі та впроваджено в навчальний процес для використання здобувачам освіти за спеціальністю «Комп'ютерні науки». Інноваційні методи навчання допомагають здобувачам освіти самостійно ліквідувати прогалини у знаннях без використання додаткового аудиторного часу чи викладача.

навчальний тренажер, освітня діяльність, середовище розробки програм NetBeans IDE, мова програмування Java

Одержано (Received) 16.03.2023

Прорецензовано (Reviewed) 29.03.2023

Прийнято до друку (Approved) 03.04.2023