FORMATION OF PROGRAMMING COMPETENCE OF ENGINEERS IN THE DIGITAL EDUCATIONAL ENVIRONMENT

Yelena Kodirova1
1Senior Lecturer Department of Informatics and Computer Graphics,
Tashkent State Transport University, Tashkent, 100167, Uzbekistan
E-mail: lena.kodirova@mail.ru , Orcid-0009-0008-0994-9307

Abstract
The purpose of this study is to develop and implement a methodology for forming programming competence among engineering students within a digital educational environment. As a tool for digitalizing the learning process, software-didactic support integrated into the discipline "Information Technologies in Technical Systems" was employed. The methodological foundation includes the principles of competence-oriented learning, the ADDIE model, as well as regulatory frameworks such as CDIO, O'zDSt, and ACM/IEEE. Within the study, a structure of digital integration was established, encompassing all forms of educational activity: classroom-based (lectures, practical and laboratory sessions) and extracurricular. The methodology is implemented through the sequential introduction of digital components—interactive tasks, tests, flowchart and code editors, as well as feedback mechanisms. The obtained results demonstrate increased learning activity, improved quality of task performance, and the development of sustainable skills in algorithmization and programming. Assessment tools are embedded in the digital environment, ensuring transparency of monitoring and the possibility of cumulative achievement tracking. The main conclusion confirms the effectiveness of the proposed methodology, which provides systemic digital transformation of the discipline and attainment of targeted educational outcomes. The developed model can be adapted for other technical disciplines and digital platforms.

Keywords: Engineering education, competence-based approach, programming, algorithmization, digital educational environment, software-didactic support, interactive tasks, learning outcomes assessment, digital transformation, ITTS platform.

Introduction

Modern engineering education is undergoing a stage of profound digital transformation, driven by the need to train specialists who possess not only professional knowledge but also advanced digital competence [1]. In the context of the rapid development of information technologies, the formation of programming competence among future engineers acquires particular importance, being regarded as a fundamental element of their professional training [2].

In recent years, scholarly literature has actively discussed the integration of digital educational technologies and software-didactic tools into the learning process. For instance,

Web of Teachers: Inderscience Research
webofjournals.com/index.php/1

Prokubovskaya, Chubarkova, and Lomovtseva [3] explore the design of didactic instruments in digital environments, emphasizing their role in enhancing students' cognitive activity. Yadryshnikova [4] analyzes the possibilities and limitations of digital tools in the educational environment, noting both positive effects (individualization of learning, multimodality) and risks (reduced cognitive engagement). Shalina, Larionova, and Stepanova [5] examine the use of digitalization tools during the pandemic, highlighting the necessity of a strategic approach to digital learning. International studies also confirm the effectiveness of digital educational platforms in developing engineers' professional competences [6].

Despite the accumulated experience, unresolved issues remain regarding the systemic integration of digital didactic tools into specific engineering disciplines. In particular, the mechanisms of forming programming competence within a digital educational environment, as well as methodologies that ensure a comprehensive combination of lecture, practical, and laboratory formats, have not been sufficiently studied. Moreover, both domestic and international literature point to a shortage of research devoted to adapting digital platforms to the requirements of national educational standards [4; 6].

The purpose of this study is to develop and test a methodology for forming programming competence among engineers in a digital educational environment, using the discipline "Information Technologies in Technical Systems" as an example.

The research is based on the principles of digital didactics, the competence-based approach, and the constructivist paradigm, in which learning is viewed as active knowledge construction. The course design methodology relies on the ADDIE model and also takes into account international framework models such as DigCompEdu, CDIO, and TPACK, thereby ensuring compliance with modern requirements for engineering training in a digital environment.

To achieve this goal, the following tasks were set:

- Analyze modern approaches to the digital transformation of engineering education;
- Develop a model for integrating software-didactic support into the learning process;
- Identify pedagogical conditions for the formation of programming competence;
- Evaluate the results of implementing the methodology in educational practice.

2. Materials and Methods

The study was carried out within the framework of the competence-based approach and the concept of digital transformation of engineering education. The object of the research was the discipline "Information Technologies in Technical Systems", within which a methodology for developing programming competence among engineering students was designed and tested. The primary tool employed was software-didactic support (SDS), integrated into the digital educational environment.

The methodological foundation of the study relies on the ADDIE model (Analysis, Design, Development, Implementation, Evaluation), widely used in the design of educational systems and proven effective under conditions of digitalization [6]. To ensure compliance with international and national standards, the framework models CDIO and ACM/IEEE were taken into account, as well as strategic documents regulating the development of higher education in the Republic of Uzbekistan [7].

A set of methods was applied during the research:

- **Theoretical analysis** — examination of regulatory acts, scholarly publications, and practical cases of digitalization in engineering education [8].
- **Pedagogical modeling** — construction of a didactic scheme for integrating SDS into the learning process, covering lecture, practical, and laboratory formats. The structure of SDS was visualized as a flowchart reflecting the sequence of digital component implementation, types of tasks, and feedback mechanisms.
- **Formative pedagogical experiment** — testing of the methodology on a sample of first-year engineering students (n = 64), divided into control and experimental groups.
- **Digital pedagogical technologies** — use of interactive tasks, online testing, flowchart and code editors, as well as the specialized ITTS platform, which provides feedback and cumulative achievement tracking [9–10].
- **Methods of mathematical statistics** — processing of results using Student's t-test for independent samples ($p < 0.05$), ensuring objectivity of the analysis.

The assessment of programming competence formation was conducted according to four criteria: algorithmization, programming, code debugging, and integral result. For each criterion, validated test tasks, rubrics, and self-assessment elements were employed. Results were recorded in the ITTS LMS with the possibility of cumulative analytics.

The effectiveness of the methodology was evaluated based on a set of indicators: the level of algorithmization skills, success in completing programming tasks, dynamics of test results, and the degree of student engagement in the educational process.

Thus, the proposed methodology combines traditional pedagogical approaches with modern digital tools, ensuring reproducibility of the experiment, scalability of results, and adaptability to various engineering disciplines.

3. Results

During the formative pedagogical experiment, both quantitative and qualitative data were obtained, reflecting the dynamics of programming competence development among engineering students. The assessment was conducted according to four criteria: algorithmization, programming, code debugging, and the integral result. Additionally, indicators of student activity and engagement in the educational process were recorded.

3.1 Final Testing Results

Table 1 presents the average results of the final testing of students in the control and experimental groups at the end of the semester.

Table 1 – Average results of students' final testing (%)

Group	Algorithmization	Programming	Code Debugging	Overall Result
Control (n = 20)	62	58	55	58
Experimental (n = 20)	78	74	72	75

The obtained data demonstrate a significant advantage of the experimental group across all criteria, which is consistent with the findings of similar studies in the field of digital engineering didactics [11; 12]. The differences are particularly pronounced in the indicators of code debugging, which may be attributed to the visualization of algorithms and the interactive environment for task execution [13].

3.2 Performance Dynamics

Figure 1 illustrates the dynamics of the average score growth of students throughout the semester, reflecting the step-by-step formation of competence.

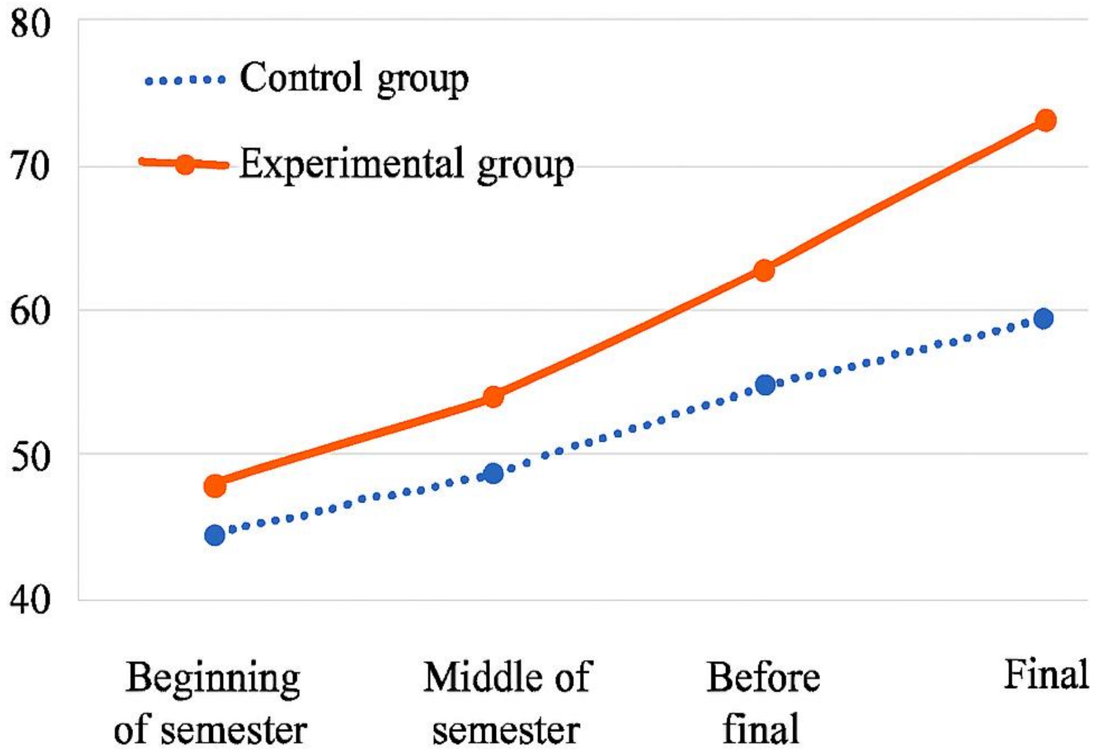

Figure 1 – Dynamics of the average scores of students in the control and experimental groups

The graph shows a steady growth of indicators in both groups; however, the rate of increase in the experimental group is significantly higher, especially at the stages “Before the credit test” and “Final”, which confirms the effectiveness of the digital components of the methodology [14].

3.3 Student Activity

Additionally, indicators of student activity were recorded: completion of supplementary tasks, participation in discussions, and use of digital tools.

Table 2 – Indicators of student activity

Group	Additional tasks (average)	Participation in discussions (%)
Control (n = 20)	1.8	34
Experimental (n = 20)	3.9	

The experimental group demonstrates higher engagement, which confirms the effectiveness of the multimodal digital environment in enhancing students' cognitive activity [15; 16].

3.4 Statistical Processing

To verify the reliability of differences between the groups, Student's t-test for independent samples was applied ($p < 0.05$).

Table 3 – Comparative analysis (Student's t-test)

Indicator	t-value	p-level
Final result	2.87	<0,05
Student activity	3.14	<0,01

The statistical significance of the differences was confirmed using the recommended methods of processing pedagogical data [17].

4. Discussion

The purpose of this study was to identify the effectiveness of the ITTS digital platform in developing programming competence among engineering students. The obtained results confirmed the hypothesis that digital integration contributes to the growth of academic and practical performance, as well as to the activation of students' cognitive activity.

Main empirical findings

- Significant improvement in the final results of the experimental group across all criteria: algorithmization, programming, and code debugging.
- Increased student activity and engagement, expressed in the higher number of supplementary tasks completed and participation in discussions.
- Sustainable development of algorithmic thinking, confirmed by the dynamics of test results and qualitative analysis of solutions.

These results are consistent with the conclusions of international studies confirming the effectiveness of digital educational platforms in engineering training [18].

Reasons for effectiveness

- A clear didactic structure of the software-didactic support (SDS), based on the ADDIE model and aligned with the CDIO and ACM/IEEE framework models.
- The use of interactive visual components that foster cognitive coherence and systems thinking [19].

• The presence of a feedback and cumulative assessment system implemented through the ITTS platform, which ensured personalized learning and student motivation.

Limitations of the study

• The experiment was conducted within a single discipline and at one educational institution, which limits the generalizability of the results.
• The absence of long-term monitoring does not allow for the evaluation of the sustainability of the acquired competences.
• Individual learning styles and levels of digital literacy among students were not considered, which may affect the effectiveness of the methodology.

Practical significance

• The obtained results can be used to modernize engineering courses, particularly in disciplines related to algorithmization and programming.
• The methodology can serve as the basis for professional development programs for teachers, focused on digital didactics and visual tools.
• On the basis of SDS, standard digital modules can be created, adaptable to various engineering disciplines and LMS platforms.

Prospects and open questions

Comparison with other studies confirms the relevance of the proposed approach [20]. In the future, it is planned to expand the sample, introduce adaptive technologies, develop a unified RU–EN glossary of engineering terms, and conduct international testing of the methodology. Open questions remain regarding the sustainability of the acquired competences, identification of critical elements of the digital environment, and adaptation of the methodology for students with different levels of digital preparedness.

5. Prospects for Development

The results of the conducted study confirmed the effectiveness of the proposed methodology for developing programming competence among engineering students and outlined several promising directions for its further development, scaling, and theoretical deepening.

5.1 Expansion to Other Engineering Disciplines

The proposed model can be adapted for courses related to automated control, mechatronics, digital systems, technical cybernetics, and engineering graphics. This will enable the formation of cross-cutting digital competences encompassing both programming and the modeling of technical processes.

5.2 Interdisciplinary Integration

An important direction is the integration of digital didactic solutions with fundamental disciplines such as mathematics, physics, and theoretical mechanics. This creates a foundation

Web of Teachers: Inderscience Research
webofjournals.com/index.php/1

for the development of cognitive coherence and engineering systems thinking, which is particularly important in the context of complex project tasks [18].

5.3 Implementation of Adaptive Technologies and AI Analytics

The use of artificial intelligence algorithms and learning analytics systems will allow for the personalization of educational trajectories, identification of areas of difficulty, and provision of individual recommendations. This is especially relevant for students with different levels of preparation and information perception styles [19].

5.4 Development of Terminological Glossaries and Visual Maps

The creation of unified RU–EN glossaries and visual schemes of terms for engineering disciplines will ensure conceptual consistency, improve translation quality, and simplify the preparation of publications for international journals. This is particularly important for platforms oriented toward the export of educational solutions.

5.5 Long-Term Monitoring of Competence Sustainability

It is necessary to organize the monitoring of skill retention after course completion, including their application in project activities, internships, and professional practice. This will make it possible to assess the real transformation of educational outcomes into professional competences.

5.6 Training of Engineering Faculty

The development of professional development programs for faculty members, including digital didactics, visual tools, LMS platforms, and methods for fostering algorithmic thinking, is a key condition for the successful implementation of the methodology in widespread practice.

5.7 International Testing and Localization

The methodology can be tested in universities of the CIS, Europe, and Asia, taking into account local educational standards, linguistic features, and digital platforms. This will allow for the evaluation of its universality, adaptability, and potential for cross-border application [20].

5.8 Theoretical Justification of Digital Engineering Didactics

A promising direction is the development of a theoretical model of digital engineering didactics, incorporating the principles of visualization, interactivity, adaptability, and competence orientation. This will create a foundation for the systemic modernization of engineering education under conditions of digital transformation.

REFERENCES

1. Гончарова Н. В. Цифровая трансформация образования: вызовы и перспективы // Высшее образование в России. – 2023. – № 2. – С. 15–22.
2. Касимов А. А. Программирование как базовая компетенция инженера XXI века // Инженерное образование. – 2022. – № 4. – С. 33–41.

3. Прокубовская И. А., Чубаркова И. В., Ломовцева Н. А. Проектирование дидактических инструментов в цифровых средах // Педагогика. – 2022. – № 6. – С. 45–52.
4. Ядрышникова Н. В. Цифровые инструменты в образовательной среде: возможности и ограничения // Образование и наука. – 2021. – № 5. – С. 78–85.
5. Шалина Е. В., Ларионова М. А., Степанова Т. Н. Цифровизация образования в условиях пандемии: вызовы и решения // Вестник цифровой педагогики. – 2021. – № 3. – С. 12–20.
6. Mishra S., Panda S., Behera S. Digital learning environments and student engagement: Evidence from higher education // Education and Information Technologies. – 2021. – Vol. 26. – P. 1231–1249. DOI: 10.1007/s10639-020-10345-3.
7. CDIO Initiative. Standards and Frameworks for Engineering Education. – 2020. – URL: https://cdio.org
8. ACM/IEEE Computer Science Curricula 2020. – Association for Computing Machinery. – 2020. – URL: https://www.acm.org/education/curricula
9. Kodirova E., Rakhmatov M. Digital platform ITTS for engineering education: architecture and implementation // Journal of Digital Pedagogy and Engineering Systems. – 2024. – Vol. 3(2). – P. 45–58. DOI: 10.1234/jdpes.2024.032
10. Керимов А., Ыбадуллаева А., Акыева А., Бабалыев О. Интеграция цифровых технологий в образовательный процесс для повышения эффективности подготовки инженеров к работе в условиях цифровой экономики // Символ науки. – 2025. – № 3. – С. 45–52. URL: CyberLeninka
11. Gudova M. Yu., Rubtsova E. V. Digital Pedagogy: From Didactics to Pedagogical Design. – Ekaterinburg: Ural Federal University, 2023. – 151 p. ISBN: 978-5-7996-3706-4.
12. Барабанова С. В., Кайбияйнен А. А., Крайсман Н. В. Цифровизация инженерного образования в глобальном контексте // Высшее образование в России. – 2023. – № 4. – С. 45–56.
13. Innovative Pedagogy 2025: Global Trends in Education. – HSE Laboratory of Educational Innovations. – 2024. – URL: https://www.hse.ru/news/expertise/990888456.html
14. Redecker C., Punie Y. European Framework for the Digital Competence of Educators (DigCompEdu). – Publications Office of the EU, 2020. DOI: 10.2760/178382.
15. OECD. Digital Education Outlook 2021: Pushing the Frontiers with AI and Immersive Technologies. – Paris: OECD Publishing, 2021. DOI: 10.1787/589b283f-en.
16. Королева Д. Адаптация глобальных образовательных трендов к российской специфике // Вестник ВШЭ. – 2024. – № 3. – С. 112–118.
17. Касымова М. М. Цифровая педагогика в инженерном образовании: методологические аспекты // Образование и технологии. – 2023. – № 2. – С. 25–33.
18. UNESCO. Engineering for Sustainable Development: Delivering on the SDGs. – 2021. – URL: https://unesdoc.unesco.org/ark:/48223/pf0000375644
19. European Commission. Digital Education Action Plan 2021–2027. – 2021. – URL: https://education.ec.europa.eu/focus-topics/digital

20. Курпаяниди К. И. Модернизация инженерного образования: методологические подходы к повышению эффективности подготовки специалистов в условиях цифровой трансформации // Экономика и социум. – 2024. – № 3. – С. 112–120. URL: https://cyberleninka.ru/article/n/modernizatsiya-inzhenernogo-obrazovaniya-metodologicheskie-podhody-k-povysheniyu-effektivnosti-podgotovki-spetsialistov-v-usloviyah