

КОМПЬЮТЕРНЫЕ ИНСТРУМЕНТЫ В ОБРАЗОВАНИИ

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ «ЛЭТИ»

№ 2, 2025

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

Абрамович Сергей Михайлович	кандидат физ.-мат. наук, профессор (США)
Бруно Бухбергер	PhD, профессор (Австрия)
Адлай Семён Франкович	кандидат физ.-мат. наук, научный сотрудник (Россия)
Бутиков Евгений Иванович	кандидат физ.-мат. наук, профессор (Россия)
Вакуленко Сергей Августович	доктор физ.-мат. наук, (Россия)
Васильев Владимир Николаевич	доктор техн. наук, член-корреспондент РАН (Россия)
Стивен Вольфрам	PhD, основатель Wolfram Research (США)
Гнитецкая Татьяна Николаевна	доктор пед. наук, профессор (Россия)
Дима Григорьев	доктор физ.-мат. наук (Франция)
Евгений Зима	PhD, профессор (Канада)
Куприянов Михаил Степанович	доктор техн. наук, профессор (Россия)
Матиясевич Юрий Владимирович	доктор физ.-мат. наук, академик РАН, советник РАН (Россия)
Новиков Борис Асенович	доктор физ.-мат. наук, профессор (Россия)
Новиков Фёдор Александрович	доктор техн. наук (Россия)
Павлов Дмитрий Алексеевич	кандидат физ.-мат. наук, инженер-исследователь (Россия)
Панков Павел Сергеевич	доктор физ.-мат. наук, профессор, член-корр. НАН (Киргизия)
Парфёнов Владимир Глебович	доктор техн. наук, профессор (Россия)
Посов Илья Александрович	кандидат техн. наук (Россия)
Семёнов Алексей Львович	доктор физ.-мат. наук, академик РАН, академик РАО, проф. (Россия)
Снегурова Виктория Игоревна	доктор пед. наук (Россия)
Сергей Соловьёв	кандидат физ.-мат. наук, Dr. habil., профессор (Франция)
Марк Спиваковский	директор по исследованиям (Франция)
Стафеев Сергей Константинович	доктор техн. наук, профессор (Россия)
Степанов Алексей Владимирович	доктор физ.-мат. наук, доцент (Россия)
Терехов Андрей Николаевич	доктор физ.-мат. наук, профессор (Россия)
Тискин Александр Владимирович	кандидат физ.-мат. наук, доцент (Россия)
Стивен Уотт	PhD, профессор (Канада)
Чирцов Александр Сергеевич	кандидат физ.-мат. наук, доктор техн. наук, профессор (Россия)

ГЛАВНЫЙ РЕДАКТОР:

Поздняков Сергей Николаевич доктор пед. наук, доцент, профессор СПбГЭТУ «ЛЭТИ»

СОРЕДАКТОР:

Васильев Николай Николаевич кандидат физ.-мат. наук, ПОМИ РАН, профессор СПбГЭТУ «ЛЭТИ»

РЕДАКТОР АНГЛИЙСКОЙ ВЕРСИИ:

Алкивиадис Акритас профессор университета Фессалии (Греция)

Журнал входит в перечень ведущих рецензируемых научных журналов и изданий ВАК по следующим специальностям:

Технические науки: 1.2.2; 2.3.5; 2.3.8. Физико-математические науки: 2.3.5; 1.1.5; 1.1.6; 1.2.3.

COMPUTER TOOLS IN EDUCATION

MINISTRY OF EDUCATION AND SCIENCE OF THE RUSSIAN FEDERATION
SAINT-PETERSBURG STATE ELECTROTECHNICAL UNIVERSITY «LETI»

№ 2, 2025

EDITORIAL BOARD:

Sergei Abramovich	<i>Candidate Sc., Professor (USA)</i>
Bruno Buchberger	<i>PhD, Professor (Austria)</i>
Semjon F. Adlaj	<i>Candidate Sc., Researcher (Russia)</i>
Eugene I. Butikov	<i>Candidate Sc., Professor (Russia)</i>
Aleksander S. Chirtsov	<i>Doctor Sc., Candidate Sc., Professor (Russia)</i>
Tatyana N. Gnitetskaya	<i>Doctor Sc., Professor (Russia)</i>
Dima Grigoriev	<i>Doctor Sc., Professor (France)</i>
Mikhail S. Kupriyanov	<i>Doctor Sc., Professor (Russia)</i>
Yury V. Matiyasevich	<i>Doctor Sc., Academician, Councilor RAS (Russia)</i>
Boris A. Novikov	<i>Doctor Sc., Professor (Russia)</i>
Fedor A. Novikov	<i>Doctor Sc., Professor (Russia)</i>
Dmitry A. Pavlov	<i>Candidate Sc., Research Engineer (Russia)</i>
Pavel S. Pankov	<i>Doctor Sc., Professor, Corr. Member NAS (Kyrgyzstan)</i>
Vladimir G. Parfenov	<i>Doctor Sc., Professor (Russia)</i>
Ilya A. Posov	<i>Candidate Sc. (Russia)</i>
Aleksej L. Semenov	<i>Doctor Sc., Professor, Academician RAS, RAE (Russia)</i>
Viktoria I. Snegurova	<i>Doctor Sc., Professor (Russia)</i>
Sergei Soloviev	<i>Doctor Sc., Dr. Habil., Professor (France)</i>
Mark Spivakovsky	<i>PhD, Research Director (France)</i>
Sergei K. Stafeev	<i>Doctor Sc., Professor (Russia)</i>
Aleksei V. Stepanov	<i>Doctor Sc., Docent (Russia)</i>
Andrej N. Terehov	<i>Doctor Sc., Professor (Russia)</i>
Alexander V. Tiskin	<i>Candidate Sc., Docent (Russia)</i>
Sergei A. Vakulenko	<i>Doctor Sc., Professor (Russia)</i>
Vladimir N. Vasilyev	<i>Doctor Sc., Professor, Corr. Member RAS (Russia)</i>
Stephen Watt	<i>PhD, Professor (Canada)</i>
Stephen Wolfram	<i>PhD, founder Wolfram Research (USA)</i>
Eugene Zima	<i>PhD, Professor (Canada)</i>

EDITOR-IN-CHIEF:

Sergei N. Pozdniakov *Professor at St. Petersburg Electrotechnical University LETI (Russian Federation)*

CO-EDITOR:

Nikolai N. Vassiliev *Senior Research Fellow at St. Petersburg Department of V. A. Steklov Mathematical Institute, Professor at St. Petersburg Electrotechnical University LETI (Russian Federation)*

EDITOR OF ENGLISH PART OF THE JOURNAL:

Alkiviadis G. Akritas *Prof., University of the Thessaly (Greece)*

The journal is included in the list of leading peer-reviewed scientific journals and publications, in which the basic scientific results of dissertations must be published for obtaining the academic degrees of doctor and candidate of sciences (list of VAK)

СОДЕРЖАНИЕ

АЛГОРИТМИЧЕСКАЯ МАТЕМАТИКА И МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

- Кривулин Н. К.,
Яковлев Д. М. Решение многокритериальных задач оценки альтернатив
на основе парных сравнений. Часть II 5

ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И МАШИННОЕ ОБУЧЕНИЕ

- Калмыков М. А.,
Шичкина Ю. А. Сравнение методов генерации синтетических нестационарных
ЭКГ-подобных сигналов для тестирования алгоритмов
анализа временных рядов 24
- Ивашенко А. О.,
Вяткин А. А.,
Бушмелёв Ф. В.,
Абрамов М. В. Моделирование системы профориентации
с использованием анализа данных социальных сетей 36

ИНЖЕНЕРИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

- Мамонов А. А.,
Салпагаров С. И.,
Матюшкин Д. В.,
Миронов Д. А.,
Кройтор О. К. Разработка системы автоматизированной подготовки документов
с использованием библиотеки Apache POI 48

КОМПЬЮТЕР В УЧЕБНОМ ПРОЦЕССЕ

- Багоутдинова А. Г. Создание и применение заданий формата «пропуски» в онлайн-курсах
на платформе Stepik 59
- Винокурова Д. В. Автоматизация выбора начальных приближений для решения
нелинейных уравнений численными методами 73

ПРАКТИКА ПРОГРАММИРОВАНИЯ

- Половикова О. Н.,
Смолякова Л. Л. Соревнования по программированию на платформе Яндекс.Контест:
кому нужны и как создавать 81

Редактор: А. А. Посова
Вёрстка: Т. Г. Мяндр
Дизайн обложки: А. И. Баранов

*Журнал издается при поддержке
НИИ информационных технологий Санкт-Петербургского государственного университета*

Все права защищены. Никакая часть данного издания не может быть воспроизведена в какой бы то ни было форме без письменного разрешения редакции журнала. Рукописи принимаются только в электронном виде. Приглашаем авторов к сотрудничеству.

Адрес редакции и издателя:
197022 Санкт-Петербург, ул. Профессора Попова, д. 5, литера Ф. Тел. +7 921 7439234
URL: <http://cte.eltech.ru/ojs> E-mail: info@kio.spb.ru

Журнал основан в 1997 г.
Издание перерегистрировано Федеральной службой по надзору за соблюдением законодательства
в сфере массовых коммуникаций и охраны культурного наследия (г. Москва)
Свидетельство ПИ № ФС 77-63337 от 09 октября 2015 г.

Учредитель:
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)
Тираж: 100 экз. Свободная цена

Подписано в печать: 20.08.2025.

C O N T E N T

ALGORITHMIC MATHEMATICS AND MATHEMATICAL MODELING

N. K. Krivulin, D. M. Yakovlev	Solving Multicriteria Problems of Rating Alternatives Based on Pairwise Comparisons. Part II	5
-----------------------------------	---	---

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

M. A. Kalmykov, Y. A. Shichkina	Comparison of Methods for Generating Synthetic Non-Stationary ECG-Like Signals for Testing Time Series Analysis Algorithms	24
A. O. Ivashchenko, A. A. Vyatkin, F. V. Bushmelev, M. V. Abramov	Modelling a Career Guidance System Using Social Network Data Analysis	36

SOFTWARE ENGINEERING

A. A. Mamonov, S. I. Salpagarov, D. V. Matyushkin, D. A. Mironov, O. K. Kroytor	Development of an Automated Document Preparation System Using the Apache POI Library	48
---	---	----

COMPUTER IN EDUCATION

A. G. Bagoutdinova	Developing and Using Fill-in-the-Blank Tasks for Online Courses on the StepiK Platform	59
D. V. Vinokurova	Automation of Initial Approximations Choice for Solving Nonlinear Equations by Numerical Methods	73

PROGRAMMING PRACTICE

O. N. Polovikova, L. L. Smolyakova	Programming Competitions on the Yandex.Contest Platform: Who Needs Them and How to Create Them	81
---------------------------------------	---	----

Editor: A. Posova
Pagination: T. Myand
Cover design: A. Baranov

*The journal is published with the support of the
Scientific Research Institute of Information Technologies of St. Petersburg State University*

All rights reserved. No part of this publication may be reproduced in any form without the written permission of the publisher. Manuscripts are accepted only in electronic form. We ask the authors to cooperate.

Editorial and publisher address:
197022 Russia, Saint Petersburg, str. Professor Popov, 5, lit. F. Tel. +7 921 7439234
URL: <http://cte.eltech.ru/ojs> E-mail: info@kio.spb.ru

The journal was founded in 1997.
The publication is registered by the Federal Service for the Supervision of Compliance of Legislation
in Mass Communications and Protection of Cultural Heritage (Moscow)
Certificate number ПИ № ФС77-63337, October 9, 2015
Founder: Saint Petersburg Electrotechnical University «LETI»
Free distribution

Signed print: 20-08-2025.



РЕШЕНИЕ МНОГОКРИТЕРИАЛЬНЫХ ЗАДАЧ ОЦЕНКИ АЛЬТЕРНАТИВ НА ОСНОВЕ ПАРНЫХ СРАВНЕНИЙ. Часть II

Кривулин Н. К.¹, доктор физ.-мат. наук, профессор, ✉ nkk@math.spbu.ru,
orcid.org/0000-0003-3070-9355

Яковлев Д. М.¹, студент, denis.yakovlev03@bk.ru

¹ Санкт-Петербургский государственный университет,
Университетская наб., д. 7–9, 199034, Санкт-Петербург, Россия

Аннотация

Рассматривается ряд известных в литературе многокритериальных задач оценки альтернатив на основе парных сравнений. В этих задачах, исходя из заданных матриц, содержащих результаты парных сравнений критериев и альтернатив, необходимо найти абсолютный рейтинг (приоритет, вес) каждой альтернативы при принятии решений. Представлены решения задач, полученные с помощью метода анализа иерархий, метода взвешенных геометрических средних, а также метода лог-чебышевской аппроксимации матриц парных сравнений. Полученные результаты показывают, что для некоторых задач решения, найденные разными методами, могут существенно отличаться друг от друга. Принятие решения о выборе наилучшей альтернативы в таких случаях может опираться на дополнительный анализ и сопоставление результатов решения задачи, которые были получены всеми применяемыми методами.

Ключевые слова: многокритериальные задачи принятия решений, парные сравнения, метод анализа иерархий, тропическая математика.

Цитирование: Кривулин Н. К., Яковлев Д. М. Решение многокритериальных задач оценки альтернатив на основе парных сравнений. Часть II // Компьютерные инструменты в образовании. 2025. № 2. С. 5–23. doi:10.32603/2071-2340-2025-2-5-23

1. ВВЕДЕНИЕ

Многокритериальные задачи оценки альтернатив на основе парных сравнений [1–3] образуют класс проблем в области принятия решений, которые представляют значительный теоретический интерес и имеют большое прикладное значение. В таких задачах на основе результатов парных сравнений альтернатив по нескольким неравнозначным критериям, а также парных сравнений самих критериев, требуется построить оценки предпочтения (рейтинги, приоритеты, веса) альтернатив для выбора наиболее подходящей альтернативы при принятии решений. Полученные результаты могут быть использованы для ранжирования альтернатив в соответствии с их оценками.

Для решения многокритериальных задач парных сравнений используется целый ряд методов, включая эвристический метод анализа иерархий [2, 4, 5] и формальный метод

взвешенных геометрических средних [6–8]. В то же время известно, что существующие методы решения могут приводить к различным и даже противоположным результатам. Поэтому проблема разработки новых методов, которые способны дополнить и расширить имеющиеся инструменты решения, остается достаточно актуальной.

Одним из методов, которые предлагают альтернативные подходы к решению многокритериальных задач парных сравнений, является метод на основе лог-чебышевской аппроксимации матриц парных сравнений, предложенный и изученный в работах [9–11]. Нахождение рейтингов альтернатив сводится к решению ряда задач оптимизации в терминах тропической алгебры, которая изучает теорию и приложения алгебраических систем с идемпотентными операциями [12–16]. Указанный подход позволяет получить решение задачи в аналитическом виде в компактной векторной форме.

В настоящей статье рассматривается ряд известных в литературе многокритериальных задач парных сравнений из работ [17–20]. Приведены решения этих задач при помощи методов анализа иерархий и взвешенных геометрических средних, а также метода лог-чебышевской аппроксимации. Статья является продолжением работ [21, 22], которые представляют результаты, полученные студентами математико-механического факультета СПбГУ при выполнении индивидуальных проектов по курсу «Принятие решений».

2. МНОГОКРИТЕРИАЛЬНАЯ ЗАДАЧА ПАРНЫХ СРАВНЕНИЙ

Пусть требуется оценить рейтинги N альтернатив $\mathcal{A}_1, \dots, \mathcal{A}_N$ на основе их парных сравнений в соответствии с K критериями. Известны результаты парных сравнений критериев, которые образуют матрицу $C = (c_{kl})$ порядка K , где величина $c_{kl} > 0$ показывает во сколько раз критерий k более значим при оценке альтернатив, чем критерий l . Имеются также результаты парных сравнений альтернатив по каждому критерию в форме матриц $A_k = (a_{ij}^{(k)})$ порядка N , где $a_{ij}^{(k)} > 0$ показывает во сколько раз альтернатива \mathcal{A}_i предпочтительнее альтернативы \mathcal{A}_j по критерию k . Многокритериальная задача парных сравнений состоит в том, чтобы с помощью матриц парных сравнений C, A_1, \dots, A_K построить вектор $x = (x_i)$ порядка N , где $x_i > 0$ определяет абсолютный рейтинг альтернативы \mathcal{A}_i при принятии решения о выборе наилучшей альтернативы.

Для решения задачи парных сравнений могут использоваться различные методы, включая наиболее распространенные — метод анализа иерархий и метод взвешенных геометрических средних, а также новый метод лог-чебышевской аппроксимации, которые применяются в настоящей работе. С целью полноты изложения ниже приводятся краткие описания и расчетные формулы для перечисленных методов (см. также [21]).

2.1. Метод анализа иерархий

Метод анализа иерархий [2, 4, 5] представляет собой эвристический метод решения многокритериальных задач парных сравнений, который получил наибольшее распространение на практике. Обозначим через $w = (w_1, \dots, w_K)^T$ нормированный относительно суммы элементов главный собственный вектор матрицы C , а через x_k нормированный главный собственный вектор матрицы A_k для каждого $k = 1, \dots, K$. Тогда вектор рейтингов альтернатив определяется как взвешенная сумма по формуле

$$x = \sum_{k=1}^K w_k x_k. \quad (1)$$

2.2. Метод взвешенных геометрических средних

Метод взвешенных геометрических средних [6–8] является формальным методом решения, в котором результат получают с помощью log-евклидовой аппроксимации матриц парных сравнений. Пусть $w = (w_1, \dots, w_K)^T$ обозначает нормированный относительно суммы координат вектор геометрических средних элементов строк матрицы C .

В качестве решения задачи парных сравнений берется (обычно после нормирования) вектор $x = (x_i)$ с элементами, которые находятся по формуле

$$x_i = \prod_{k=1}^K \left(\prod_{j=1}^N a_{ij}^{(k)} \right)^{w_k / N}, \quad i = 1, \dots, N. \quad (2)$$

2.3. Метод log-чебышевской аппроксимации

Метод log-чебышевской аппроксимации опирается на решение задач аппроксимации матриц парных сравнений в метрике Чебышева в логарифмической шкале [9–11]. Решение задачи log-чебышевской аппроксимации обычными методами оказывается слишком сложным и на практике обычно не применяется. В то же время, аналитическое решение задачи может быть получено в терминах тропической алгебры, которая изучает теорию и приложения алгебраических систем с идемпотентными операциями [12–16]. Операция называется идемпотентной, если результат ее применения к операндам с одинаковым значением равен этому значению (например, идемпотентной является операция вычисления максимума, для которой справедливо равенство $\max(x, x) = x$).

Решение, полученное с помощью log-чебышевской аппроксимации, может быть неединственным и предлагать разные оценки альтернатив. Чтобы упростить выбор альтернатив в случае неединственного решения в работах [9, 10] используется подход на основе определения в некотором смысле «наилучшего» и «наихудшего» решений. При этом выбирается наилучший дифференцирующий вектор рейтингов альтернатив, для которого отношение между максимальным и минимальным элементами максимально, и наихудший дифференцирующий вектор, для которого это отношение минимально.

2.3.1. Мах-алгебра

Решение многокритериальной задачи парных сравнений на основе log-чебышевской аппроксимации записывается в терминах алгебраической системы, которая обычно называется мах-алгеброй. В этой системе для всех неотрицательных вещественных чисел задана операция сложения, которая обозначается символом \oplus и определяется как вычисление максимума так, что $x \oplus y = \max(x, y)$. Операция умножения обозначается и определяется как обычно, а при записи выражений знак операции умножения \times опускается.

Операции над матрицами и векторами с элементами из множества неотрицательных вещественных чисел выполняется поэлементно по обычным правилам с заменой арифметического сложения $+$ на операцию \oplus . Нулевая и единичная матрицы, а также нулевой вектор имеют обычный вид. Единичная матрица обозначается символом I .

Для квадратной матрицы $A = (a_{ij})$ порядка N след вычисляется по формуле

$$\text{tr } A = a_{11} \oplus a_{22} \oplus \dots \oplus a_{NN} = \bigoplus_{i=1}^N a_{ii}.$$

Спектральный радиус матрицы A находится как сумма

$$\lambda = \text{tr } A \oplus \text{tr}^{1/2}(A^2) \oplus \dots \oplus \text{tr}^{1/N}(A^N) = \bigoplus_{i=1}^N \text{tr}^{1/i}(A^i).$$

При условии $\lambda \leq 1$ для матрицы A определена матрица Клини в виде

$$A^* = I \oplus A \oplus \dots \oplus A^{N-1} = \bigoplus_{i=0}^{N-1} A^i.$$

Для любого ненулевого вектора-столбца $x = (x_i)$ определен сопряженный вектор-строка $x^- = (x_i^-)$ с элементами $x_i^- = x_i^{-1}$ если $x_i \neq 0$, и $x_i^- = 0$ в противном случае.

Норма вектора $x = (x_i)$ порядка N вычисляется по формуле

$$\|x\| = x_1 \oplus x_2 \oplus \dots \oplus x_N = \bigoplus_{i=1}^N x_i.$$

2.3.2. Решение многокритериальной задачи парных сравнений

В этом разделе описывается вычислительная процедура решения задачи парных сравнений на основе лог-чебышевской аппроксимации с применением тропической алгебры [9–11]. Процедура включает следующие шаги, где все выражения записаны и должны вычисляться в терминах тах-алгебры с операцией сложения \oplus , которая выполняется как вычисление максимума.

1. Определение для матрицы C парных сравнений критериев наилучшего и наихудшего дифференцирующих векторов весов.

- 1.1. Построение генерирующей матрицы D для векторов весов критериев:

$$D = (\lambda^{-1}C)^* = \bigoplus_{k=0}^{K-1} (\lambda^{-1}C)^k, \quad \lambda = \bigoplus_{k=1}^K \text{tr}^{1/k}(C^k). \quad (3)$$

- 1.2. Вычисление по матрице D со столбцами d_1, \dots, d_K наилучшего вектора весов как наименьшего (при покомпонентном сравнении) на множестве векторов:

$$w = d_l \|d_l\|^{-1}, \quad l = \arg \max_{1 \leq k \leq K} \|d_k\| \|d_k^-\|. \quad (4)$$

Если наименьший вектор не единственный, то в качестве наилучших берутся все векторы, которые не доминируют над каким-либо другим вектором.

- 1.3. Вычисление по матрице D наихудшего дифференцирующего вектора весов:

$$v = (1^T D)^-. \quad (5)$$

2. Определение для матриц A_1, \dots, A_K парных сравнений альтернатив и вектора весов $w = (w_k)$ наилучшего дифференцирующего вектора рейтингов альтернатив.

- 2.1. Вычисление взвешенной суммы матриц парных сравнений альтернатив:

$$P = \bigoplus_{k=1}^K w_k A_k. \quad (6)$$

- 2.2. Построение генерирующей матрицы Q для векторов рейтингов альтернатив:

$$Q = (\mu^{-1}P)^* = \bigoplus_{n=0}^{N-1} (\mu^{-1}P)^n, \quad \mu = \bigoplus_{n=1}^N \text{tr}^{1/n}(P^n). \quad (7)$$

- 2.3. Вычисление по матрице Q со столбцами q_1, \dots, q_N наилучшего вектора рейтингов альтернатив как наименьшего на множестве векторов:

$$x = q_m \|q_m\|^{-1}, \quad m = \arg \max_{1 \leq n \leq N} \|q_n\| \|q_n^-\|. \quad (8)$$

Если наименьший вектор не единственный, то в качестве наилучших берутся все векторы, которые не доминируют над каким-либо другим вектором.

3. Определение для матриц A_1, \dots, A_K парных сравнений альтернатив и вектора весов $v = (v_k)$ наилучшего дифференцирующего вектора рейтингов альтернатив.

- 3.1. Вычисление взвешенной суммы матриц парных сравнений альтернатив:

$$R = \bigoplus_{k=1}^K v_k A_k. \quad (9)$$

- 3.2. Построение генерирующей матрицы S для векторов рейтингов альтернатив:

$$S = (v^{-1} R)^* = \bigoplus_{n=0}^{N-1} (v^{-1} R)^n, \quad v = \bigoplus_{n=1}^N \text{tr}^{1/n}(R^n). \quad (10)$$

- 3.3. Вычисление по матрице S наилучшего дифференцирующего вектора рейтингов альтернатив:

$$y = (1^T S)^-. \quad (11)$$

3. ПРИМЕРЫ МНОГОКРИТЕРИАЛЬНЫХ ЗАДАЧ ПАРНЫХ СРАВНЕНИЙ

В этом разделе рассматриваются многокритериальные задачи оценки альтернатив на основе парных сравнений, которые изучались в работах [17–20]. Представлены примеры решения этих задач с помощью методов анализа иерархий, взвешенных геометрических средних и log-чебышевской аппроксимации.

Сначала приведено достаточно подробное решение для задачи небольшой размерности с тремя альтернативами и четырьмя критериями. Для других задач, имеющих более высокую размерность, решения представлены в сокращенной форме.

3.1. Выбор автомобиля

Рассмотрим задачу выбора автомобиля из $N = 3$ альтернатив в соответствии с $K = 4$ критериями, описанную в [17]. Критерии сравнения автомобилей включают:

- 1) престижность,
- 2) стоимость,
- 3) экономичность расхода топлива,
- 4) уровень комфорта.

Матрица парных сравнений критериев задана в виде

$$C = \begin{pmatrix} 1 & 1/4 & 1/3 & 1/2 \\ 4 & 1 & 3 & 3/2 \\ 3 & 1/3 & 1 & 1/3 \\ 2 & 2/3 & 3 & 1 \end{pmatrix}.$$

Сравниваются автомобили следующих марок: Acura TL, Toyota Camry и Honda Civic. Результаты парных сравнений по каждому из критериев составляют матрицы

$$A_1 = \begin{pmatrix} 1 & 8 & 4 \\ 1/8 & 1 & 1/4 \\ 1/4 & 4 & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & 1/4 & 1/9 \\ 4 & 1 & 1/5 \\ 9 & 5 & 1 \end{pmatrix}, \quad A_3 = \begin{pmatrix} 1 & 2/3 & 1/3 \\ 3/2 & 1 & 1/2 \\ 3 & 2 & 1 \end{pmatrix}, \quad A_4 = \begin{pmatrix} 1 & 4 & 7 \\ 1/4 & 1 & 3 \\ 1/7 & 1/3 & 1 \end{pmatrix}.$$

3.1.1. Метод анализа иерархий

Решение с помощью метода анализа иерархий требует нахождения нормированных относительно суммы элементов главных собственных векторов для всех матриц парных сравнений. Для матрицы C парных сравнений критериев имеем собственный вектор

$$w \approx (0,0987 \quad 0,4250 \quad 0,1686 \quad 0,3078)^T.$$

Собственные векторы матриц парных сравнений A_1 , A_2 , A_3 и A_4 записываются в виде

$$x_1 \approx \begin{pmatrix} 0,7071 \\ 0,0702 \\ 0,2227 \end{pmatrix}, \quad x_2 \approx \begin{pmatrix} 0,0633 \\ 0,1939 \\ 0,7429 \end{pmatrix}, \quad x_3 \approx \begin{pmatrix} 0,1818 \\ 0,2727 \\ 0,5455 \end{pmatrix}, \quad x_4 \approx \begin{pmatrix} 0,7049 \\ 0,2109 \\ 0,0841 \end{pmatrix}.$$

После вычисления вектора рейтингов альтернатив по формуле (1), нормирования относительно максимального элемента и определения порядка альтернатив, получим

$$x = w_1 x_1 + w_2 x_2 + w_3 x_3 + w_4 x_4 \approx \begin{pmatrix} 0,3443 \\ 0,2002 \\ 0,4555 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 0,7558 \\ 0,4395 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 > \mathcal{A}_1 > \mathcal{A}_2.$$

3.1.2. Метод взвешенных геометрических средних

Для применения метода взвешенных геометрических средних найдем векторы геометрических средних для элементов строк матриц парных сравнений. Матрица парных сравнений критериев C имеет нормированный относительно суммы элементов вектор геометрических средних в форме

$$w \approx (0,0964 \quad 0,4396 \quad 0,1622 \quad 0,3018)^T.$$

Вычисляя векторы геометрических средних для A_1 , A_2 , A_3 и A_4 , получим

$$x_1 \approx \begin{pmatrix} 3,1748 \\ 0,3150 \\ 1 \end{pmatrix}, \quad x_2 \approx \begin{pmatrix} 0,3029 \\ 0,9283 \\ 3,5569 \end{pmatrix}, \quad x_3 \approx \begin{pmatrix} 0,6057 \\ 0,9086 \\ 1,8171 \end{pmatrix}, \quad x_4 \approx \begin{pmatrix} 3,0366 \\ 0,9086 \\ 0,3625 \end{pmatrix}.$$

Вектор рейтингов альтернатив, найденный по формуле (2), результат его нормирования по максимальному элементу и порядок альтернатив записываются в виде

$$x = \begin{pmatrix} x_{11}^{w_1} x_{12}^{w_2} x_{13}^{w_3} x_{14}^{w_4} \\ x_{21}^{w_1} x_{22}^{w_2} x_{23}^{w_3} x_{24}^{w_4} \\ x_{31}^{w_1} x_{32}^{w_2} x_{33}^{w_3} x_{34}^{w_4} \end{pmatrix} \approx \begin{pmatrix} 0,8523 \\ 0,8281 \\ 1,4167 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 0,6016 \\ 0,5845 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 > \mathcal{A}_1 > \mathcal{A}_2.$$

3.1.3. Метод log-чебышевской аппроксимации

В этом разделе описывается последовательность расчетов согласно процедуре, приведенной в разделе 2.3.2, где все вычисления выполняются в терминах тах-алгебры, для которой операция сложения \oplus выполняется как тах, а умножения — как обычно.

Для того, чтобы определить веса критериев, необходимо найти степени матрицы парных сравнений критериев C . В результате последовательных вычислений получим

$$C^2 = \begin{pmatrix} 1 & 1/3 & 3/2 & 1/2 \\ 9 & 1 & 9/2 & 2 \\ 3 & 3/4 & 1 & 3/2 \\ 9 & 1 & 3 & 1 \end{pmatrix}, \quad C^3 = \begin{pmatrix} 9/2 & 1/2 & 3/2 & 1/2 \\ 27/2 & 9/4 & 6 & 9/2 \\ 3 & 1 & 9/2 & 3/2 \\ 9 & 9/4 & 3 & 9/2 \end{pmatrix}, \quad C^4 = \begin{pmatrix} 9/2 & 9/8 & 3/2 & 9/4 \\ 18 & 27/8 & 27/2 & 27/4 \\ 27/2 & 3/2 & 9/2 & 3/2 \\ 9 & 3 & 27/2 & 9/2 \end{pmatrix}.$$

С помощью формул (3) определим спектральный радиус матрицы C в виде

$$\lambda = \text{tr} C \oplus \text{tr}^{1/2}(C^2) \oplus \text{tr}^{1/3}(C^3) \oplus \text{tr}^{1/4}(C^4) = 2^{-1/3} 3^{2/3} \approx 1,6510.$$

Затем составим матрицу $\lambda^{-1}C$ и найдем для нее матрицу Клини

$$D = (\lambda^{-1}C)^* = \begin{pmatrix} 1 & 1/4\lambda & \lambda/3 & 1/2\lambda \\ 2\lambda & 1 & 3/\lambda & 1 \\ 3/\lambda & \lambda/6 & 1 & \lambda/3 \\ 2\lambda & 1/2 & 3/\lambda & 1 \end{pmatrix}.$$

Чтобы определить наилучший дифференцирующий вектор весов сначала вычислим

$$\|d_1\| \|d_1^-\| = \|d_3\| \|d_3^-\| = \|d_4\| \|d_4^-\| = 2\lambda \approx 3,3019, \quad \|d_2\| \|d_2^-\| = 4\lambda \approx 6,6039.$$

Учитывая, что условию в (4) удовлетворяет только второй столбец, находим наилучший вектор весов в виде

$$w = d_2 \|d_2\|^{-1} = (1/4\lambda \quad 1 \quad \lambda/6 \quad 1/2)^T.$$

Вычисление наихудшего дифференцирующего вектора весов по формуле (5) дает

$$v = (1^T D)^- = (1/2\lambda \quad 1 \quad \lambda/3 \quad 1)^T.$$

Составим взвешенную сумму (6) матриц парных сравнений альтернатив

$$P = w_1 A_1 \oplus w_2 A_2 \oplus w_3 A_3 \oplus w_4 A_4 = \begin{pmatrix} 1 & 2 & 7/2 \\ 4 & 1 & 3/2 \\ 9 & 5 & 1 \end{pmatrix}.$$

Используя формулы (7), сначала для матрицы P найдем степени

$$P^2 = \begin{pmatrix} 63/2 & 35/2 & 7/2 \\ 27/2 & 8 & 14 \\ 20 & 18 & 63/2 \end{pmatrix}, \quad P^3 = \begin{pmatrix} 70 & 63 & 441/4 \\ 126 & 70 & 189/4 \\ 567/2 & 315/2 & 70 \end{pmatrix},$$

а затем вычислим ее спектральный радиус

$$\mu = \text{tr} P \oplus \text{tr}^{1/2}(P^2) \oplus \text{tr}^{1/3}(P^3) = 2^{-1/2} 3 \cdot 7^{1/2} \approx 5,6125.$$

Составим матрицу $\mu^{-1}P$ и построим матрицу Клини

$$Q = (\mu^{-1}P)^* = \begin{pmatrix} 1 & 5/9 & \mu/9 \\ 4/\mu & 1 & 4/9 \\ 2\mu/7 & 5/\mu & 1 \end{pmatrix}.$$

Чтобы определить наилучший дифференцирующий вектор альтернатив по формулам (8), найдем

$$\|q_1\| \|q_1^-\| = \|q_3\| \|q_3^-\| = 9/4, \quad \|q_2\| \|q_2^-\| = 9/5.$$

Условию в (8) удовлетворяют первый и третий столбцы, которые приводят к одному наилучшему вектору и соответствующему ему упорядочению альтернатив в форме

$$x = q_1 \|q_1\|^{-1} = q_3 \|q_3\|^{-1} = \begin{pmatrix} \mu/9 \\ 4/9 \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0,6363 \\ 0,4444 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 > \mathcal{A}_1 > \mathcal{A}_2.$$

Теперь рассмотрим взвешенную сумму (9), которая принимает вид

$$R = v_1 A_1 \oplus v_2 A_2 \oplus v_3 A_3 \oplus v_4 A_4 = \begin{pmatrix} 1 & 4 & 7 \\ 4 & 1 & 3 \\ 9 & 5 & 1 \end{pmatrix}.$$

По формулам (10) для матрицы R найдем степени

$$R^2 = \begin{pmatrix} 63 & 35 & 12 \\ 27 & 16 & 28 \\ 20 & 36 & 63 \end{pmatrix}, \quad R^3 = \begin{pmatrix} 140 & 252 & 441 \\ 252 & 140 & 189 \\ 567 & 315 & 140 \end{pmatrix}$$

и определим спектральный радиус

$$\nu = \text{tr } R \oplus \text{tr}^{1/2}(R^2) \oplus \text{tr}^{1/3}(R^3) = 3 \cdot 7^{1/2} \approx 7,9373.$$

Для матрицы $\nu^{-1}R$ найдем матрицу Клини

$$S = (\nu^{-1}R)^* = \begin{pmatrix} 1 & 5/9 & \nu/9 \\ 4/\nu & 1 & 4/9 \\ \nu/7 & 5/\nu & 1 \end{pmatrix}.$$

В результате вычисления по формуле (11) наихудшего дифференцирующего вектора рейтингов альтернатив и определения порядка альтернатив получим:

$$y = (\mathbf{1}^T S)^- = \begin{pmatrix} 7/\nu \\ 1 \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0,8819 \\ 1 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 \equiv \mathcal{A}_2 > \mathcal{A}_1.$$

Заметим, что все полученные векторы рейтингов альтернатив указывают на третью альтернативу (Honda Civic) как наиболее предпочтительную. Решения с помощью методов анализа иерархий и взвешенных геометрических средних, а также наилучшее дифференцирующее решение метода лог-чебышевской аппроксимации устанавливают один и тот же порядок альтернатив. Упорядочение на основе наихудшего дифференцирующего решения меняет местами первую и вторую альтернативы (Acura TL и Toyota Camry).

3.2. Выбор дома

Найдем решение для задачи из работы [18] о выборе из $N = 3$ альтернатив при покупке дома, которые сравниваются по $K = 8$ критериям. Выбраны следующие критерии:

- 1) размер дома (кладовая, количество комнат, площадь комнат, общая площадь дома),
- 2) общественный транспорт (расположение вблизи автобусных остановок),
- 3) городской район (небольшое автомобильное движение, безопасность, красивый вид, низкие налоги, общее состояние района),
- 4) возраст дома,
- 5) состояние двора (размеры двора, расстояние до соседей),
- 6) бытовая техника (посудомоечная машина, измельчитель мусора, кондиционер, сигнализация и т. п.),
- 7) общее состояние (стены, ковровые покрытия, шторы, электропроводка),
- 8) возможности финансирования (ипотека, скидки продавца, банка).

Результаты парных сравнений критериев составляют матрицу

$$C = \begin{pmatrix} 1 & 5 & 3 & 7 & 6 & 6 & 1/3 & 1/4 \\ 1/5 & 1 & 1/3 & 5 & 3 & 3 & 1/5 & 1/7 \\ 1/3 & 3 & 1 & 6 & 3 & 4 & 6 & 1/5 \\ 1/7 & 1/5 & 1/6 & 1 & 1/3 & 1/4 & 1/7 & 1/8 \\ 1/6 & 1/3 & 1/3 & 3 & 1 & 1/2 & 1/5 & 1/6 \\ 1/6 & 1/3 & 1/4 & 4 & 2 & 1 & 1/5 & 1/6 \\ 3 & 5 & 1/6 & 7 & 5 & 5 & 1 & 1/2 \\ 4 & 7 & 5 & 8 & 6 & 6 & 2 & 1 \end{pmatrix}.$$

Матрицы парных сравнений альтернатив в соответствии с критериями имеют вид

$$\begin{aligned} A_1 &= \begin{pmatrix} 1 & 6 & 8 \\ 1/6 & 1 & 4 \\ 1/8 & 1/4 & 1 \end{pmatrix}, & A_2 &= \begin{pmatrix} 1 & 7 & 1/5 \\ 1/7 & 1 & 1/8 \\ 5 & 8 & 1 \end{pmatrix}, & A_3 &= \begin{pmatrix} 1 & 8 & 6 \\ 1/8 & 1 & 1/4 \\ 1/6 & 4 & 1 \end{pmatrix}, \\ A_4 &= \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, & A_5 &= \begin{pmatrix} 1 & 5 & 4 \\ 1/5 & 1 & 1/3 \\ 1/4 & 3 & 1 \end{pmatrix}, & A_6 &= \begin{pmatrix} 1 & 8 & 6 \\ 1/8 & 1 & 1/5 \\ 1/6 & 5 & 1 \end{pmatrix}, \\ A_7 &= \begin{pmatrix} 1 & 1/2 & 1/2 \\ 2 & 1 & 1 \\ 2 & 1 & 1 \end{pmatrix}, & A_8 &= \begin{pmatrix} 1 & 1/7 & 1/5 \\ 7 & 1 & 3 \\ 5 & 1/3 & 1 \end{pmatrix}. \end{aligned}$$

3.2.1. Метод анализа иерархий

Чтобы найти решение задачи, вычисляется нормированный главный собственный вектор матрицы C и нормированные главные собственные векторы матриц A_1, \dots, A_8 . Затем по формуле (1) определяется вектор рейтингов альтернатив. Полученный вектор рейтингов альтернатив, нормированный вектор рейтингов и соответствующий порядок альтернатив записываются в виде

$$x \approx \begin{pmatrix} 0,3959 \\ 0,3407 \\ 0,2634 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 1 \\ 0,8606 \\ 0,6652 \end{pmatrix}, \quad A_1 > A_2 > A_3.$$

3.2.2. Метод взвешенных геометрических средних

После вычисления нормированного вектора геометрических средних для строк матрицы C и векторов геометрических средних для матриц A_1, \dots, A_8 вектор рейтингов альтернатив находится по формуле (2). Определение вектора рейтингов, нормированного вектора рейтингов и порядка альтернатив дает следующий результат:

$$x \approx \begin{pmatrix} 1,0234 \\ 1,0302 \\ 0,9485 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 0,9933 \\ 1 \\ 0,9207 \end{pmatrix}, \quad \mathcal{A}_2 > \mathcal{A}_1 > \mathcal{A}_3.$$

3.2.3. Метод log-чебышевской аппроксимации

Для определения вектора весов критериев сначала по формулам (3) вычисляется спектральный радиус λ матрицы D , составляется матрица $\lambda^{-1}C$ и находится матрица Клини $D = (\lambda^{-1}C)^*$. Полученные результаты записываются в виде

$$\lambda = 3 \cdot 2^{1/3} \approx 3,7798, \quad D = \begin{pmatrix} 1 & 5/3 & 3/\lambda & 7/3 & 5/3 & 5/3 & \lambda/3 & 1/6 \\ 1/9 & 1 & 1/3\lambda & 5/\lambda & 3/\lambda & 3/\lambda & \lambda/27 & 5\lambda/432 \\ \lambda/3 & 5\lambda/9 & 1 & 7\lambda/9 & 5\lambda/9 & 5\lambda/9 & 6/\lambda & \lambda/18 \\ 1/18 & 5/54 & 1/6\lambda & 1 & 5/54 & 5/54 & \lambda/54 & 1/8\lambda \\ 1/9 & 5/27 & 1/3\lambda & 3/\lambda & 1 & 5/27 & \lambda/27 & 1/6\lambda \\ 1/12 & 5/36 & 1/4\lambda & 4/\lambda & 2/\lambda & 1 & \lambda/36 & 1/6\lambda \\ 3/\lambda & 5/\lambda & \lambda/6 & 7/\lambda & 5/\lambda & 5/\lambda & 1 & 1/2\lambda \\ 5/3 & 25/9 & 5/\lambda & 35/9 & 25/9 & 25/9 & 5\lambda/9 & 1 \end{pmatrix}.$$

Наилучший дифференцирующий вектор весов устанавливается по формулам (4):

$$w = d_8 \|d_8\|^{-1} = (1/6 \quad 5\lambda/432 \quad \lambda/18 \quad 1/8\lambda \quad 1/6\lambda \quad 1/6\lambda \quad 1/2\lambda \quad 1)^T.$$

Вычисление наихудшего дифференцирующего вектора весов по формуле (5) дает:

$$v = (3/5 \quad 9/25 \quad \lambda/5 \quad 9/35 \quad 9/25 \quad 9/25 \quad 9/5\lambda \quad 1)^T.$$

Определение взвешенной суммы матриц парных сравнений альтернатив по формуле (6) для нахождения наилучшего дифференцирующего вектора рейтингов альтернатив приводит к следующему результату:

$$P = \begin{pmatrix} 1 & 4\lambda/9 & 4/3 \\ 7 & 1 & 3 \\ 5 & 2\lambda/9 & 1 \end{pmatrix}.$$

После вычисления по формулам (7) спектрального радиуса μ матрицы P и определения матрицы Клини $Q = (\mu^{-1}P)^*$ получим

$$\mu = 2^{7/6} 3^{-1/2} 7^{1/2} \approx 3,4292, \quad Q = \begin{pmatrix} 1 & \mu/7 & 3/7 \\ 7/\mu & 1 & 3/\mu \\ 5/\mu & 5/7 & 1 \end{pmatrix}.$$

Найдем наилучший дифференцирующий вектор рейтингов альтернатив с помощью формул (8). Получим вектор рейтингов и порядок альтернатив в виде

$$x = q_3 \|q_3\|^{-1} = \begin{pmatrix} 3/7 \\ 3/\mu \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0,4286 \\ 0,8748 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 > \mathcal{A}_2 > \mathcal{A}_1.$$

Для нахождения наихудшего дифференцирующего вектора в соответствии с (9) составим взвешенную сумму матриц

$$R = \begin{pmatrix} 1 & 8\lambda/5 & 24/5 \\ 7 & 1 & 3 \\ 5 & 4\lambda/5 & 1 \end{pmatrix}.$$

По формулам (10) найдем спектральный радиус ν матрицы R и построим матрицу Клини $S = (\nu^{-1}R)^*$. В результате получим

$$\nu = 2^{5/3} 3^{1/2} 5^{-1/2} 7^{1/2} \approx 6,5064, \quad S = \begin{pmatrix} 1 & \nu/7 & \nu/7 \\ 7/\nu & 1 & 1 \\ 1 & \nu/7 & 1 \end{pmatrix}.$$

Используя формулу (11), находим наихудший дифференцирующий вектор рейтингов и соответствующий порядок альтернатив в виде

$$y = \begin{pmatrix} \nu/7 \\ 1 \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0,9295 \\ 1 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 \equiv \mathcal{A}_2 > \mathcal{A}_1.$$

В этой задаче применяемые методы решения задачи парных сравнений по-разному определяют наиболее предпочтительную альтернативу. Учитывая, что вторая альтернатива чаще других ставится на первое или второе место, эту альтернативу можно рекомендовать для выбора в качестве возможного решения.

3.3. Выбор компании для приобретения

Рассмотрим задачу из работы [19] о выборе компании для приобретения (поглощения) из $N = 3$ альтернатив. При оценке компаний используются $K = 7$ критериев:

- 1) финансовые показатели,
- 2) потенциал роста,
- 3) климат трудовых отношений,
- 4) конкурентоспособность,
- 5) организационная культура,
- 6) размер компании,
- 7) отраслевая общность.

Матрица парных сравнений критериев имеет вид

$$C = \begin{pmatrix} 1 & 3 & 7 & 5 & 1 & 7 & 1 \\ 1/3 & 1 & 9 & 1 & 1 & 5 & 1 \\ 1/7 & 1/9 & 1 & 1/7 & 1/5 & 1/2 & 1/4 \\ 1/5 & 1 & 7 & 1 & 1/4 & 7 & 1/3 \\ 1 & 1 & 5 & 4 & 1 & 5 & 3 \\ 1/7 & 1/5 & 2 & 1/7 & 1/5 & 1 & 1/6 \\ 1 & 1 & 4 & 3 & 1/3 & 6 & 1 \end{pmatrix}.$$

Результаты сравнений альтернатив по каждому критерию образуют матрицы

$$A_1 = \begin{pmatrix} 1 & 9 & 3 \\ 1/9 & 1 & 1/5 \\ 1/3 & 5 & 1 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 1 & 7 & 4 \\ 1/7 & 1 & 1/3 \\ 1/4 & 3 & 1 \end{pmatrix}, \quad A_3 = \begin{pmatrix} 1 & 1/5 & 1/3 \\ 5 & 1 & 2 \\ 3 & 1/2 & 1 \end{pmatrix},$$

$$A_4 = \begin{pmatrix} 1 & 6 & 3 \\ 1/6 & 1 & 1/2 \\ 1/3 & 2 & 1 \end{pmatrix}, \quad A_5 = \begin{pmatrix} 1 & 1/9 & 1/5 \\ 9 & 1 & 4 \\ 5 & 1/4 & 1 \end{pmatrix}, \quad A_6 = \begin{pmatrix} 1 & 1/7 & 1/4 \\ 7 & 1 & 3 \\ 4 & 1/3 & 1 \end{pmatrix},$$

$$A_7 = \begin{pmatrix} 1 & 1/7 & 1/3 \\ 7 & 1 & 3 \\ 3 & 1/3 & 1 \end{pmatrix}.$$

3.3.1. Метод анализа иерархий

Применение метода анализа иерархий приводит к следующим результатам:

$$x \approx \begin{pmatrix} 0,3946 \\ 0,3640 \\ 0,2414 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 1 \\ 0,9225 \\ 0,6118 \end{pmatrix}, \quad A_1 > A_2 > A_3.$$

3.3.2. Метод взвешенных геометрических средних

Метод взвешенных геометрических средних приводит к решению в виде

$$x \approx \begin{pmatrix} 1,0391 \\ 0,9227 \\ 1,0430 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 0,9963 \\ 0,8847 \\ 1 \end{pmatrix}, \quad A_3 > A_2 > A_1.$$

3.3.3. Метод лог-чебышевской аппроксимации

В результате вычисления спектрального радиуса матрицы C получим

$$\lambda = 2^{1/5} 5^{-1/5} 7^{1/5} 9^{1/5} \approx 1,9067.$$

Построим матрицу Клини $D = (\lambda^{-1}C)^*$ в виде

$$D = \begin{pmatrix} 1 & 3/\lambda & 25\lambda^2/9 & 5/\lambda & 5\lambda/9 & 35/\lambda^2 & 5/3 \\ 5\lambda^2/42 & 1 & 5 & 5\lambda^2/14 & 1/\lambda & 5\lambda/2 & 3/\lambda^2 \\ \lambda^2/42 & \lambda/14 & 1 & \lambda^2/14 & 1/5\lambda & \lambda/2 & 3/5\lambda^2 \\ 1/3 & 1/\lambda & 14/\lambda^2 & 1 & \lambda^2/9 & 7/\lambda & \lambda/3 \\ 3/\lambda^2 & 5\lambda^2/14 & 5\lambda & 9/\lambda^2 & 1 & 5\lambda^2/2 & 3/\lambda \\ \lambda/21 & 1/7 & 2/\lambda & \lambda/7 & 2/5/\lambda^2 & 1 & \lambda^2/21 \\ 1/\lambda & 3/\lambda^2 & 5\lambda^2/3 & 3/\lambda & \lambda/3 & 21/\lambda^2 & 1 \end{pmatrix}.$$

В качестве наилучшего дифференцирующего вектора весов берется

$$w = d_1 \|d_1\|^{-1} = (1 \quad 5\lambda^2/42 \quad \lambda^2/42 \quad 1/3 \quad 3/\lambda^2 \quad \lambda/21 \quad 1/\lambda)^T.$$

Вычисляя наихудший дифференцирующий вектор весов, получим

$$v = (1 \quad \lambda/3 \quad 9/25\lambda^2 \quad \lambda/5 \quad 9/5\lambda \quad \lambda^2/35 \quad 3/5)^T.$$

С помощью вектора w составим взвешенную сумму матриц сравнений альтернатив

$$P = \begin{pmatrix} 1 & 9 & 3 \\ 27/\lambda^2 & 1 & 12/\lambda^2 \\ 15/\lambda^2 & 5 & 1 \end{pmatrix}.$$

Для матрицы P найдем спектральный радиус и построим матрицу Клини

$$\mu = 2^{-1/5} 3^{21/10} 5^{1/5} 7^{-1/5} \approx 8,1757, \quad Q = \begin{pmatrix} 1 & 9/\mu & 4/9 \\ \mu/9 & 1 & 4\mu/81 \\ 5/9 & 5/\mu & 1 \end{pmatrix}.$$

Определение наилучшего дифференцирующего вектора рейтингов альтернатив дает

$$x = q_3 \|q_3\|^{-1} = \begin{pmatrix} 4/9 \\ 4\mu/81 \\ 1 \end{pmatrix} \approx \begin{pmatrix} 0,4444 \\ 0,4037 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 > \mathcal{A}_1 > \mathcal{A}_2.$$

Используя вектор v , составим взвешенную сумму матриц сравнений альтернатив

$$R = \begin{pmatrix} 1 & 9 & 3 \\ 81/5\lambda & 1 & 36/5\lambda \\ 9/\lambda & 5 & 1 \end{pmatrix}.$$

В результате вычисления спектрального радиуса для R и матрицы Клини имеем

$$\nu = 2^{-1/10} 3^{14/5} 5^{-2/5} 7^{-1/10} \approx 8,7446, \quad S = \begin{pmatrix} 1 & 9/\nu & 4/9 \\ \nu/9 & 1 & 4\nu/81 \\ 5/9 & 5/\nu & 1 \end{pmatrix}.$$

Наихудший дифференцирующий вектор рейтингов и порядок альтернатив записываются в следующем виде:

$$y = \begin{pmatrix} 1 \\ \nu/9 \\ 9/4 \end{pmatrix} \approx \begin{pmatrix} 1 \\ 0,9716 \\ 1 \end{pmatrix}, \quad \mathcal{A}_3 \equiv \mathcal{A}_1 > \mathcal{A}_2.$$

Учитывая, что два метода (метод взвешенных геометрических средних и метод логчебышевской аппроксимации) из трех приписывают наибольший рейтинг третьей альтернативе, эту альтернативу можно рекомендовать как наиболее предпочтительную. Заметим, что такое решение будет противоречить результату применения метода анализа иерархий, который присваивает наибольший рейтинг первой альтернативе.

3.4. Выбор высшего учебного заведения

В этом разделе рассматривается задача, представленная в [20], о выборе из $N = 5$ альтернатив высших учебных заведений, которые сравниваются по $K = 6$ критериям:

- 1) академическая среда и автономия,
- 2) статус и чувство принадлежности,
- 3) долгосрочные перспективы,
- 4) наставничество студентов,
- 5) открытость взаимодействия,
- 6) поддержка и сопровождение.

Результаты парных сравнений критериев составляют матрицу

$$C = \begin{pmatrix} 1 & 2 & 3 & 6 & 6 & 5 \\ 1/2 & 1 & 3 & 6 & 6 & 5 \\ 1/3 & 1/3 & 1 & 4 & 4 & 3 \\ 1/6 & 1/6 & 1/4 & 1 & 2 & 1/2 \\ 1/6 & 1/6 & 1/4 & 1/2 & 1 & 1/4 \\ 1/5 & 1/5 & 1/3 & 2 & 4 & 1 \end{pmatrix}.$$

Матрицы парных сравнений альтернатив имеют вид

$$\begin{aligned}
 A_1 &= \begin{pmatrix} 1 & 1/3 & 1/2 & 1/6 & 2 \\ 3 & 1 & 2 & 1/2 & 4 \\ 2 & 1/2 & 1 & 1/3 & 3 \\ 6 & 2 & 3 & 1 & 7 \\ 1/2 & 1/4 & 1/3 & 1/7 & 1 \end{pmatrix}, & A_2 &= \begin{pmatrix} 1 & 6 & 3 & 2 & 7 \\ 1/6 & 1 & 1/4 & 1/2 & 3 \\ 1/3 & 4 & 1 & 1/3 & 5 \\ 1/2 & 2 & 3 & 1 & 7 \\ 1/7 & 1/3 & 1/5 & 1/7 & 1 \end{pmatrix}, \\
 A_3 &= \begin{pmatrix} 1 & 7 & 1/3 & 2 & 8 \\ 1/7 & 1 & 1/5 & 1/4 & 4 \\ 3 & 5 & 1 & 4 & 9 \\ 1/2 & 4 & 1/4 & 1 & 6 \\ 1/8 & 1/4 & 1/9 & 1/6 & 1 \end{pmatrix}, & A_4 &= \begin{pmatrix} 1 & 1/2 & 1/4 & 2 & 5 \\ 2 & 1 & 1/3 & 5 & 7 \\ 4 & 3 & 1 & 4 & 6 \\ 1/2 & 1/5 & 1/4 & 1 & 2 \\ 1/5 & 1/7 & 1/6 & 1/2 & 1 \end{pmatrix}, \\
 A_5 &= \begin{pmatrix} 1 & 1/6 & 1/8 & 2 & 3 \\ 6 & 1 & 1/4 & 5 & 7 \\ 8 & 4 & 1 & 9 & 9 \\ 1/2 & 1/5 & 1/9 & 1 & 2 \\ 1/3 & 1/7 & 1/9 & 1/2 & 1 \end{pmatrix}, & A_6 &= \begin{pmatrix} 1 & 1/5 & 1/3 & 3 & 3 \\ 5 & 1 & 5 & 6 & 6 \\ 3 & 1/5 & 1 & 2 & 2 \\ 1/3 & 1/6 & 1/2 & 1 & 2 \\ 1/3 & 1/6 & 1/2 & 1/2 & 1 \end{pmatrix}.
 \end{aligned}$$

3.4.1. Метод анализа иерархий

Вычисления в соответствии с методом анализа иерархий дают следующий результат:

$$x \approx \begin{pmatrix} 0,2235 \\ 0,1979 \\ 0,2423 \\ 0,2914 \\ 0,0450 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 0,7670 \\ 0,6790 \\ 0,8315 \\ 1 \\ 0,1546 \end{pmatrix}, \quad A_4 > A_3 > A_1 > A_2 > A_5.$$

3.4.2. Метод взвешенных геометрических средних

Решение по методу взвешенных геометрических средних имеет вид

$$x \approx \begin{pmatrix} 1,2266 \\ 1,0857 \\ 1,4519 \\ 1,6949 \\ 0,3051 \end{pmatrix}, \quad x / \max_i x_i \approx \begin{pmatrix} 0,7238 \\ 0,6406 \\ 0,8566 \\ 1 \\ 0,1800 \end{pmatrix}, \quad A_4 > A_3 > A_1 > A_2 > A_5.$$

3.4.3. Метод log-чебышевской аппроксимации

В результате вычисления спектрального радиуса матрицы C и построения матрицы Клини получим

$$\lambda = 2^{2/5} 3^{1/5} \approx 1,6438, \quad D = \begin{pmatrix} 1 & 2/\lambda & 6/\lambda^2 & 2\lambda^2 & 6\lambda & 3\lambda^2/2 \\ \lambda/2 & 1 & 3/\lambda & 12/\lambda^2 & 3\lambda^2 & 9/\lambda^2 \\ \lambda^2/6 & \lambda/3 & 1 & 4/\lambda & 12/\lambda^2 & 3/\lambda \\ 1/3\lambda^2 & \lambda^2/18 & \lambda/6 & 1 & 2/\lambda & 1/2 \\ 1/6\lambda & 1/3\lambda^2 & \lambda^2/12 & \lambda/3 & 1 & \lambda/4 \\ 2/3\lambda^2 & \lambda^2/9 & \lambda/3 & 4/3 & 4\lambda & 1 \end{pmatrix}.$$

Определение наилучшего вектора весов критериев дает следующий вектор:

$$\mathbf{w} = \mathbf{d}_1 \|\mathbf{d}_1\|^{-1} = (1 \quad \lambda/2 \quad \lambda^2/6 \quad 1/3\lambda^2 \quad 1/6\lambda \quad 2/3\lambda^2)^T.$$

Наихудший дифференцирующий вектор весов критериев записывается в виде

$$\mathbf{v} = (1 \quad \lambda/2 \quad \lambda^2/6 \quad 1/2\lambda^2 \quad 1/6\lambda \quad 2/3\lambda^2)^T.$$

После составления на основе вектора весов \mathbf{w} взвешенной суммы матриц парных сравнений альтернатив получим

$$\mathbf{P} = \begin{pmatrix} 1 & 3\lambda & 3\lambda/2 & \lambda & 7\lambda/2 \\ 3 & 1 & 2 & 4/\lambda^2 & 4 \\ 2 & 2\lambda & 1 & 2\lambda^2/3 & 5\lambda/2 \\ 6 & 2 & 3 & 1 & 7 \\ 1/2 & \lambda/6 & 1/3 & 1/7 & 1 \end{pmatrix}.$$

Вычисляя спектральный радиус матрицы \mathbf{P} и матрицу Клини $\mathbf{Q} = (\mu^{-1}\mathbf{P})^*$, получим

$$\mu = 2^{1/5} 3^{11/10} \approx 3,8463, \quad \mathbf{Q} = \begin{pmatrix} 1 & \mu/3 & 2/3 & 108/\mu^4 & 7\mu/18 \\ 3/\mu & 1 & 2/\mu & 324/\mu^5 & 7/6 \\ 4\mu^2/81 & 4\mu^3/243 & 1 & 2\mu^3/243 & 14\mu^3/729 \\ 6/\mu & 2 & 4/\mu & 1 & 7/3 \\ 1/2\mu & 1/6 & 1/3\mu & 54/\mu^5 & 1 \end{pmatrix}.$$

Определение наилучшего дифференцирующего вектора рейтингов альтернатив приводит к следующему результату:

$$\mathbf{x} = \mathbf{q}_4 \|\mathbf{q}_4\|^{-1} = \begin{pmatrix} 108/\mu^4 \\ 324/\mu^5 \\ 2\mu^3/243 \\ 1 \\ 54/\mu^5 \end{pmatrix} \approx \begin{pmatrix} 0,4935 \\ 0,3849 \\ 0,4683 \\ 1 \\ 0,0642 \end{pmatrix}, \quad \mathcal{A}_4 > \mathcal{A}_1 > \mathcal{A}_3 > \mathcal{A}_2 > \mathcal{A}_5.$$

Вычисление взвешенной суммы на основе вектора \mathbf{v} показывает, что выполняется равенство матриц $\mathbf{R} = \mathbf{P}$. Учитывая, что тогда $\nu = \mu$ и $\mathbf{S} = \mathbf{Q}$, наихудший дифференцирующий вектор рейтингов альтернатив находится на основе матрицы \mathbf{Q} в виде

$$\mathbf{y} = \begin{pmatrix} \mu/6 \\ 1/2 \\ \mu/4 \\ 1 \\ 3/7 \end{pmatrix} \approx \begin{pmatrix} 0,6410 \\ 0,5000 \\ 0,9616 \\ 1 \\ 0,4286 \end{pmatrix}, \quad \mathcal{A}_4 > \mathcal{A}_3 > \mathcal{A}_1 > \mathcal{A}_2 > \mathcal{A}_5.$$

Для рассматриваемой задачи все решения одинаково определяют наилучшую и наихудшую альтернативы и являются в этом смысле вполне согласованными.

4. ЗАКЛЮЧЕНИЕ

Для ряда известных в литературе многокритериальных задач оценки альтернатив на основе парных сравнений приведены решения, полученные с помощью метода анализа иерархий, метода взвешенных геометрических средних, а также метода, использующего log-чебышевскую аппроксимацию матриц парных сравнений.

Показано, что упорядочение альтернатив на основе рейтингов, полученных разными методами, может различаться. Для решения проблемы выбора наилучшей альтернативы в этом случае представляется целесообразным провести анализ и сопоставление результатов применения нескольких методов. Тогда в качестве наилучшей естественно выбирать такую альтернативу, которая в результате применения этих методов получает наивысший рейтинг чаще других альтернатив.

Список литературы

1. Подиновский В. В., Ногин В. Д. Парето-оптимальные решения многокритериальных задач. М.: Наука, 1982. 256 с.
2. Саати Т. Принятие решений. Метод анализа иерархий. М.: Радио и связь, 1993. 315 с.
3. Ногин В. Д. Принятие решений в многокритериальной среде: количественный подход. М.: Физматлит, 2002. 144 с.
4. Saaty T. L. A scaling method for priorities in hierarchical structures // J. Math. Psych. 1977. Vol. 15, № 3. P. 234–281. doi:10.1016/0022-2496(77)90033-5
5. Saaty T. L. On the measurement of intangibles: A principal eigenvector approach to relative measurement derived from paired comparisons // Notices Amer. Math. Soc. 2013. Vol. 60, № 2. P. 192–208. doi:10.1090/noti944
6. Narasimhan R. A geometric averaging procedure for constructing supertransitive approximation to binary comparison matrices // Fuzzy Sets and Systems. 1982. Vol. 8, № 1. P. 53–61. doi:10.1016/0165-0114(82)90029-X
7. Crawford G., Williams C. A note on the analysis of subjective judgment matrices // J. Math. Psych. 1985. Vol. 29, № 4. P. 387–405. doi:10.1016/0022-2496(85)90002-1
8. Barzilai J., Cook W. D., Golany B. Consistent weights for judgements matrices of the relative importance of alternatives // Oper. Res. Lett. 1987. Vol. 6, № 3. P. 131–134. doi:10.1016/0167-6377(87)90026-5
9. Krivulin N., Sergeev S. Tropical implementation of the Analytical Hierarchy Process decision method // Fuzzy Sets and Systems. 2019. Vol. 377. P. 31–51. doi:10.1016/j.fss.2018.10.013.
10. Кривулин Н. К., Агеев В. А. Методы тропической оптимизации в многокритериальных задачах оценки альтернатив на основе парных сравнений // Вестн. С.-Петерб. ун-та. Прикладная математика. 2019. Т. 15, вып. 4. С. 472–488. doi:10.21638/11702/spbu10.2019.405
11. Krivulin N. Application of tropical optimization for solving multicriteria problems of pairwise comparisons using log-Chebyshev approximation // Int. J. Approx. Reason. 2024. Vol. 169. P. 109168. doi:10.1016/j.ijar.2024.109168.
12. Baccelli F. L., Cohen G., Olsder G. J., Quadrat J.-P. Synchronization and Linearity. Wiley Series in Probability and Statistics. Chichester: Wiley, 1993. 514 p.
13. Маслов В. П., Колокольцов В. Н. Идемпотентный анализ и его применение в оптимальном управлении. М.: Физматлит, 1994. 144 с.
14. Golan J. S. Semirings and Affine Equations over Them. New York: Springer, 2003. Vol. 556 of Mathematics and Its Applications. 256 p. doi:10.1007/978-94-017-0383-3
15. Heidergott B., Olsder G. J., van der Woude J. Max Plus at Work. Princeton Series in Applied Mathematics. Princeton: Princeton Univ. Press, 2006. 226 p.
16. Кривулин Н. К. Методы идемпотентной алгебры в задачах моделирования и анализа сложных систем. СПб.: Изд-во С.-Петерб. ун-та, 2009. 255 с.
17. Saaty T. L. The modern science of multicriteria decision making and its practical applications: The AHP / ANP approach // Oper. Res. 2013. Vol. 61, № 5. P. 1101–1118. doi:10.2307/24540487

18. Saaty T. L. How to make a decision: The analytic hierarchy process // European J. Oper. Res. 1990. Vol. 48, № 1. P. 9–26. doi:10.1016/0377-2217(90)90057-1
19. Bagchi P., Rao R. P. Decision making in mergers: An application of the analytic hierarchy process // Manag. Decis. Econ. 1992. Vol. 13, № 2. P. 91–99. doi:10.1002/mde.4090130202
20. Goshal D. S. K., Naskar S. K., Bose D. D. AHP in assessing performance of diploma institutes — A case study // Journal of Technical Education and Training. 2012. Vol. 3. P. 67–81.
21. Кривулин Н. К., Абильдаев Т., Горшечникова В. Д., Капаца Д., Магдич Е. А., Мандрикова А. А. О решении многокритериальных задач принятия решений на основе парных сравнений // Компьютерные инструменты в образовании. 2020. № 2. P. 27–58. doi:10.32603/2071-2340-2020-2-27-58
22. Кривулин Н. К., Булгакова Д. С., Григорьев Д. А., Нагуманова К. И., Приньков А. С., Салова Я. А., Филатова А. А. Решение многокритериальных задач оценки альтернатив на основе парных сравнений // Компьютерные инструменты в образовании. 2024. № 2. С. 5–29. doi:10.32603/2071-2340-2024-2-5-29

Поступила в редакцию 01.07.2025, окончательный вариант — 30.07.2025.

Кривулин Николай Кимович, доктор физ.-мат. наук, профессор кафедры статистического моделирования, СПбГУ, ✉ nkk@math.spbu.ru

Яковлев Денис Михайлович, студент, СПбГУ, denis.yakovlev03@bk.ru

Computer tools in education, 2025

№ 2: 5–23

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2025-2-5-23

Solving Multicriteria Problems of Rating Alternatives Based on Pairwise Comparisons. Part II

Krivulin N. K.¹, Doctor sc., Professor, ✉ nkk@math.spbu.ru, orcid.org/0000-0003-3070-9355
Yakovlev D. M.¹, Student, denis.yakovlev03@bk.ru

¹Saint Petersburg State University, 7–9 Universitetskaya emb., 199034, Saint Petersburg, Russia

Abstract

A number of well-known multicriteria problems of evaluating alternatives based on pairwise comparisons are considered. In these problems, given matrices containing results of paired comparisons of criteria and alternatives, one needs to find an absolute rating (priority, weight) of each alternative for decision making. Solutions to the problems are presented obtained using the method of analytical hierarchy process, the method of weighted geometric means, and the method of log-Chebyshev approximation of pairwise comparison matrices. The results obtained show that for some problems, solutions found by different methods may significantly differ from each other. In such cases, the decision to choose the best alternative may be based on additional analysis and comparison of the results of the problem solution obtained by all the methods used.

Keywords: multicriteria decision making problems, pairwise comparisons, analytical hierarchy process, tropical mathematics.

Citation: N. K. Krivulin and D. M. Yakovlev, "Solving Multicriteria Problems of Rating Alternatives Based on Pairwise Comparisons. Part II," *Computer tools in education*, no. 2, pp. 5–23, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-5-23

References

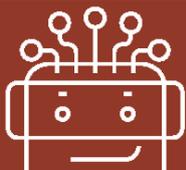
1. V. V. Podinovskii and V. D. Nogin, *Pareto-optimal'nye resheniya mnogokriterial'nykh zadach* [Pareto-optimal solutions to multicriteria problems], Moscow: Nauka, 1982 (in Russian).
2. T. Saati, *Prinyatie reshenij. Metod analiza ierarkhij* [Decision making. Hierarchy analysis method], Moscow: Radio i svyaz', 1993 (in Russian).
3. V. D. Nogin, *Prinyatie reshenii v mnogokriterial'noi srede: kolichestvennyi podkhod* [Decision making in a multicriteria environment: a quantitative approach], Moscow: Fizmatlit, 2002 (in Russian).
4. T. L. Saaty, "A scaling method for priorities in hierarchical structures," *Journal of Mathematical Psychology*, vol. 15, no. 3, pp. 234–281, 1977; doi:10.1016/0022-2496(77)90033-5
5. T. L. Saaty, "On the measurement of intangibles: A principal eigenvector approach to relative measurement derived from paired comparisons," *Notices of the American Mathematical Society*, vol. 60, no. 2, pp. 192–208, 2013; doi:10.1090/noti944
6. R. Narasimhan, "A geometric averaging procedure for constructing supertransitive approximation to binary comparison matrices," *Fuzzy Sets and Systems*, vol. 8, no. 1, pp. 53–61, 1982; doi:10.1016/0165-0114(82)90029-X
7. G. Crawford and C. Williams, "A note on the analysis of subjective judgment matrices," *Journal of Mathematical Psychology*, vol. 29, no. 4, pp. 387–405, 1985; doi:10.1016/0022-2496(85)90002-1
8. J. Barzilai, W. D. Cook, and B. Golany, "Consistent weights for judgements matrices of the relative importance of alternatives," *Operations Research Letters*, vol. 6, no. 3, pp. 131–134, 1987; doi:10.1016/0167-6377(87)90026-5
9. N. Krivulin and S. Sergeev, "Tropical implementation of the Analytical Hierarchy Process decision method," *Fuzzy Sets and Systems*, vol. 377, pp. 31–51, 2019; doi:10.1016/j.fss.2018.10.013
10. N. K. Krivulin and V. A. Ageev, "Methods of tropical optimization in multicriteria problems of rating alternatives from pairwise comparisons," *Vestnik of St Petersburg University. Applied Mathematics. Computer Science. Control Processes*, vol. 15, no. 4, pp. 472–488, 2019 (in Russian); doi:10.21638/11702/spbu10.2019.405
11. N. Krivulin, "Application of tropical optimization for solving multicriteria problems of pairwise comparisons using log-Chebyshev approximation," *International Journal of Approximate Reasoning*, vol. 169, p. 109168, 2024; doi:10.1016/j.ijar.2024.109168
12. F. L. Baccelli, G. Cohen, G. J. Olsder, and J. P. Quadrat, *Synchronization and linearity*, Wiley Series in Probability and Statistics, Chichester, UK: Wiley, 1993.
13. V. P. Maslov and V. N. Kolokol'tsov, "Idempotentnyi analiz i ego primenenie v optimal'nom upravlenii" [Idempotent analysis and its application in optimal control], Moscow: Fizmatlit, 1994 (in Russian).
14. J. S. Golan, *Semirings and affine equations over them*, Mathematics and Its Applications, vol. 556, New York: Springer, 2003; doi:10.1007/978-94-017-0383-3
15. B. Heidergott, G. J. Olsder, and J. van der Woude, *Max Plus at Work*, Princeton Series in Applied Mathematics, Princeton, NJ: Princeton Univ. Press, 2006.
16. N. K. Krivulin, *Metody idempotentnoi algebry v zadachakh modelirovaniya i analiza slozhnykh sistem* [Idempotent algebra methods in modeling and analysis of complex systems], St. Petersburg: Izdatel'stvo Sankt-Peterburgskogo universiteta, 2009.
17. T. L. Saaty, "The modern science of multicriteria decision making and its practical applications: The AHP/ANP approach," *Operations Research*, vol. 61, no. 5, pp. 1101–1118, 2019 (in Russian); doi:10.2307/24540487
18. T. L. Saaty, "How to make a decision: The analytic hierarchy process," *European Journal of Operational Research*, vol. 48, no. 1, pp. 9–26, 1990; doi:10.1016/0377-2217(90)90057-1
19. P. Bagchi and R. P. Rao, "Decision making in mergers: An application of the analytic hierarchy process," *Managerial and Decision Economics*, vol. 13, no. 2, pp. 91–99, 1992; doi:10.1002/mde.4090130202

20. D. S. K. Goshal, S. K. Naskar, and D. D. Bose, "AHP in assessing performance of diploma institutes — A case study," *Journal of Technical Education and Training*, vol. 3, pp. 67–81, 2012.
21. N. K. Krivulin, T. Abildaev, V. D. Gorshechnikova, D. Kapatsa, E. A. Magdich, and A. A. Mandrikova, "On solving multicriteria decision making problems based on pairwise comparisons," *Computer Tools in Education*, no. 2, pp. 27–58, 2020 (in Russian); doi:10.32603/2071-2340-2020-2-27-58
22. N. K. Krivulin, D. S. Bulgakova, D. A. Grigoriev, K. I. Nagumanova, A. S. Prinkov, Y. A. Salova, and A. A. Filatova, "Solving multicriteria problems of rating alternatives based on pairwise comparisons," *Computer Tools in Education*, no. 2, pp. 5–29, 2024 (in Russian); doi:10.32603/2071-2340-2024-2-5-29

Received 01-07-2025, the final version — 30-07-2025.

Nikolai Krivulin, Doctor of Sciences (Phys.-Math.), Professor, Department of Statistical Modeling, St. Petersburg State University, ✉ nkk@math.spbu.ru

Denis Yakovlev, Student, Saint Petersburg State University, denis.yakovlev03@bk.ru



СРАВНЕНИЕ МЕТОДОВ ГЕНЕРАЦИИ СИНТЕТИЧЕСКИХ НЕСТАЦИОНАРНЫХ ЭКГ-ПОДОБНЫХ СИГНАЛОВ ДЛЯ ТЕСТИРОВАНИЯ АЛГОРИТМОВ АНАЛИЗА ВРЕМЕННЫХ РЯДОВ

Калмыков М. А.¹, аспирант, ✉ mica_2011@mail.ru

Шичкина Ю. А.¹, доктор техн. наук, доцент, профессор, strange.y@mail.ru,
orcid.org/0000-0001-7140-1686

¹ Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»
им. В. И. Ульянова (Ленина), ул. Профессора Попова, 5, корп. 3, 197022, Санкт-Петербург, Россия

Аннотация

В данной работе рассмотрены различные подходы к генерации синтетических сигналов, имитирующих электрокардиограмму (ЭКГ) человека, с акцентом на нестационарность временного ряда и наличие разнообразных форм волны сигнала. Предлагаются результаты анализа трех подходов к генерации синтетических нестационарных ЭКГ-подобных сигналов, включающих: 1) правило-ориентированный подход, при котором модель ЭКГ строится на основе суммы гауссовых функций, каждая из которых моделирует характерную волну (где P — волна предсердной деполяризации, QRS — комплекс желудочковой деполяризации, T — волна реполяризации); 2) стохастические модели с использованием Марковских цепей для эмуляции переходов между различными физиологическими состояниями; 3) нейросетевые генераторы, не основанные на жестко заданных правилах (например, рекуррентная LSTM с случайными весами). Показано, как модель ЭКГ-сигнала, полученную при каждом из подходов, можно модифицировать для внесения нестационарности, в частности вариации длительности сердечных циклов, переключения состояний) и добавления локальных артефактов записи, например зашумлённых участков. Предложенные подходы могут быть использованы при тестировании алгоритмов кластеризации и анализа временных рядов, когда необходимо проверить устойчивость методов к шумам, редким событиям и смене состояний.

Ключевые слова: синтетическая ЭКГ, нестационарные сигналы, генерация сигналов, временные ряды, rule-based методы, марковские цепи, нейросетевые генераторы, LSTM.

Цитирование: Калмыков М. А., Шичкина Ю. А. Сравнение методов генерации синтетических нестационарных ЭКГ-подобных сигналов для тестирования алгоритмов анализа временных рядов // Компьютерные инструменты в образовании. 2025. № 2. С. 24–35. doi:10.32603/2071-2340-2025-2-24-35

1. ВВЕДЕНИЕ

Анализ биомедицинских временных рядов, включая электрокардиограммы (ЭКГ), являются важнейшей составляющей процесса принятия решений в современной медицине. Благодаря развитию методов машинного обучения и увеличению доступности больших массивов данных, появилась возможность более точного диагностирования и мониторинга сердечно-сосудистых заболеваний [1–4]. Однако использование реальных медицинских данных связано с рядом ограничений, включая вопросы конфиденциальности, высокие затраты на сбор и разметку данных, а также сложности, связанные с обработкой данных по редким патологиям, которые трудно найти в клинической практике [5–7].

Синтетические датасеты представляют собой эффективное решение этих проблем, так как они позволяют:

- Исключить риски, связанные с использованием персональных данных пациентов.
- Создавать специфические сценарии для тестирования алгоритмов, включая редкие патологии и артефакты записи.
- Генерировать неограниченное количество данных с контролируемыми характеристиками шума и временных изменений.

Целью настоящей работы является разработка и анализ методов генерации синтетических нестационарных сигналов, имитирующих электрокардиограмму (ЭКГ), предназначенных для создания тренировочных и тестовых наборов данных при разработке алгоритмов обработки и анализа биомедицинских временных рядов. Использование синтетических данных особенно актуально в случаях, когда реальные данные ограничены по объёму, недоступны из-за этических ограничений или не содержат достаточного количества редких патологических сценариев и артефактов. В частности, синтетические данные позволяют контролировать вариативность сигналов, включая физиологические состояния и уровень шума, и проводить систематическую оценку устойчивости алгоритмов обработки данных к изменениям условий регистрации и наличию артефактов.

За последние десятилетия проблема создания высококачественных моделей ЭКГ-сигнала привлекла значительное внимание исследователей, что нашло отражение в обширной литературе по разработке методов синтеза и анализа ЭКГ с учётом их нестационарности и физиологической изменчивости. Так, Пан Джиапу и Уиллис Томпкинс предложили алгоритм обнаружения комплекса QRS на основе реальных и синтетических данных, что позволило повысить качество детекции, однако алгоритм может демонстрировать ограничения при обработке сильно зашумлённых сигналов или нестандартных аритмий [8]. Джордж Муди и Роджер Марк разработали базу MIT-BIH Arrhythmia Database, включающую как реальные, так и синтетические записи, что стало стандартом для тестирования алгоритмов анализа ЭКГ, но база страдает недостаточным охватом редких патологических случаев [9]. Также были представлены исследования, показывающие, что добавление синтетических артефактов снижает вероятность ложных срабатываний, хотя этот подход не всегда полностью отражает сложность реальной шумовой среды [10].

Для генерации синтетических сигналов применяются различные подходы. Одним из них является подход, основанный на использовании суммы гауссиан для моделирования волн P, Q, R, S, T [11]. Этот подход позволяет задавать форму сигналов с высокой точностью, однако не учитывает нестационарность реальной ЭКГ. Другим подходом является построение стохастической модели на основе цепей Маркова. Этот подход позволяет эмулировать

вать переходы между различными формами ЭКГ-сигнала при смене физиологического состояния человека (например, при тахикардии, брадикардии) [12]. В последние годы активно развиваются нейросетевые подходы, включая генеративно-состязательные сети (GAN) и вариационные автоэнкодеры (VAE), которые позволяют обучать модели на реальных данных и синтезировать сигналы с высокой степенью реалистичности [13–15].

Несмотря на успехи в разработке методов генерации синтетических ЭКГ, остаются нерешенные проблемы. Во-первых, большинство моделей сосредоточено на стационарных сигналах, тогда как реальная ЭКГ часто изменяется во времени под воздействием различных факторов, таких как физическая нагрузка, стресс или патологии. Во-вторых, модели, обученные на реальных данных, требуют больших объёмов размеченной информации, что ограничивает их применение в условиях дефицита данных. В-третьих, генерация редких и сложных паттернов, таких как артефакты записи или внезапные изменения состояния, всё ещё остаётся сложной задачей.

Таким образом, цель данной работы — рассмотреть три различных подхода к генерации синтетических ЭКГ-подобных сигналов, акцентируя внимание на методах, позволяющих добиться нестационарности и разнообразия форм сигнала ЭКГ. Это позволит не только улучшить тестирование алгоритмов анализа временных рядов, но и выявить основные ограничения существующих подходов.

2. МЕТОДЫ И РАЗРАБОТКА СИСТЕМЫ

В данном разделе описываются три подхода к генерации нестационарных ЭКГ-подобных сигналов, условно разделяемые на: 1) правило-ориентированные, 2) стохастические (на основе Марковской цепи) и 3) «без правил» (нейросеть со случайными весами). Все подходы могут работать в режиме непрерывной генерации, при этом синтезированные данные сохраняются в файл или могут быть переданы в системы дальнейшей обработки (например, для анализа или визуализации) в режиме «реального времени».

2.1. Правило-ориентированная модель (Rule-based)

Основные требования к выходным данным модели ЭКГ-сигнала:

- фрагмент ЭКГ-сигнала представляется суммой нескольких гауссовых функций, каждая из которых моделирует типичную (стандартную) форму отдельной волны ЭКГ [6];
- параметры гауссиан (центры, ширина, амплитуда) задаются вручную, но могут слегка меняться при каждой генерации цикла.

Для разнообразия вводится несколько «кластеров» (A, B, C), где у каждого кластера свои значения параметров. Переключение между кластерами происходит с заданной вероятностью, что моделирует смену форм ЭКГ-сигнала у разных групп испытуемых или при разных состояниях.

Нестационарность синтезированного ЭКГ-сигнала достигается путём:

1. Изменения R-R интервала от цикла к циклу (например, $\pm 20\%$ относительно заданной частоты сердечных сокращений).
2. Случайной модуляции амплитуды пиков.
3. Добавления шумовых вставок (артефакты), где на коротких (или длинных) промежутках сигнал существенно зашумляется.

Данный подход крайне прост в реализации и наглядно позволяет управлять «формой» сигналов, однако он основан на заранее определённых параметрах (позициях и величинах пиков), поэтому отличается некоторой «жесткостью». Тем не менее, в практических задачах может быть полезен благодаря удобству настройки и быстроте.

2.2. Генерация с помощью Марковской цепи

Чтобы уйти от жёсткой схемы переключения кластеров, можно ввести Марковскую цепь с несколькими состояниями, например, Normal (норма), Tachy (тахикардия), Brady (брадикардия), Arrh (аритмия) [7]. Матрица перехода задаёт вероятности перехода из одного состояния в другое. Каждое состояние описывает набор параметров для генерации очередного сердечного цикла (базовая длина R-R интервала, смещения волн, амплитуд и т. п.).

Таким образом, вместо «A → B» по равновероятному выбору будет наблюдаться стохастическая эволюция физиологического состояния. Переход из «Tachy» (тахикардия) в «Arrh» (аритмия) может иметь вероятность 10 %, из «Normal» в «Arrh» — 5 % и так далее, что более правдоподобно отражает динамику состояния организма. Локальные артефакты (шумы) также можно добавлять «поверх» сгенерированного цикла. В результате получается более гибкая модель: она остаётся во многом параметрической (на базовом уровне могут продолжаться использоваться гауссовы функции или иные базовые функции), однако набор параметров при каждом новом цикле выбирается в зависимости от состояния Марковской цепи и генерируемых случайных вариаций.

2.3. Нейросетевой генератор «без правил» (Untrained LSTM)

Заметным отличием от предыдущих подходов является применение рекуррентной нейронной сети (LSTM) с инициализированными случайными весами [13]. Здесь полностью отсутствуют зафиксированные пики P, Q, R, S, T или жёстко заданные переходы между состояниями. Сеть на каждом временном шаге принимает на вход случайный шум и своё внутреннее рекуррентное состояние, формируя на выходе набор значений (амплитуд сигнала).

При этом:

- Сеть не обучена на реальных данных, поэтому не ожидается, что сигнал будет внешне похож на истинную ЭКГ.
- Однако на «выходе» всё равно образуется нелинейная временная последовательность, которую можно бесконечно генерировать в режиме «потока».
- Полное отсутствие правил демонстрирует, как выглядят сигналы, формируемые чисто случайным (хотя и рекуррентным) процессом.

Для получения таким путём реалистичной ЭКГ потребовалось бы полноценное обучение на реальном датасете, как в генеративных моделях (GAN, VAE, Diffusion models). Однако, если задача — сравнить разные способы генерации и протестировать «устойчивость» алгоритмов анализа к разнообразным шумоподобным паттернам, данный «полностью случайный» метод может оказаться полезным.

3. Шаблонно-ориентированный метод генерации синтетических ЭКГ-сигналов

В дополнение к описанным выше подходам (правило-ориентированному, стохастическому на основе марковской цепи и нейросетевому генератору «без правил»), был разра-

ботан и протестирован дополнительный метод генерации синтетических ЭКГ-сигналов, основанный на выделении шаблона из реальных данных [15]. Данный подход может быть охарактеризован как шаблонно-ориентированный (template-based).

В его основе лежит следующий алгоритм:

1. Извлечение шаблона реального сердечного цикла. Реальный сигнал подвергается детекции R пиков, после чего производится сегментация на отдельные циклы. На основании полученных циклов вычисляется средний (template) цикл, который полноценно отражает морфологию ЭКГ: характеристики волн P, QRS и T, а также временные соотношения между ними.
2. Генерация синтетических циклов. Для синтетической генерации каждый цикл формируется путём возмущения шаблонного цикла с использованием небольшого случайного множителя (noise_level в диапазоне, например, $\pm 3\%$). Дополнительно, длительность каждого синтетического цикла определяется на основе распределения реальных R-R интервалов, что обеспечивает сохранение временной динамики сигнала.
3. Объединение циклов в непрерывный сигнал. Сгенерированные циклы последовательно интерполируются до числа отсчетов, соответствующего заданной частоте дискретизации (в нашем случае — 129,92 Гц), и объединяются в единый синтетический сигнал.

Применение данного метода позволяет получить синтетические данные, в которых морфологическая форма сердечного цикла почти идентична реальной.

4. РЕЗУЛЬТАТЫ И СРАВНЕНИЕ

4.1. Примеры синтезированных сигналов

На рисунке 1 приведены 4 фрагмента (по 2–3 сердечных цикла) для каждого из рассмотренных подходов:

1. Rule-based: легко различимы пики, похожие на P, Q, R, S, T. При переходе от кластера А к В заметны изменения формы. Артефакты, если включены, выглядят как локальные всплески шумов.
2. Марковская цепь: форма волн также основана на гауссианах, но переключения между «Normal», «Tachy» и «Arrh» выглядят более естественными. При длительной генерации видна характеристическая «смена состояний» в случайные моменты времени.
3. Untrained LSTM: во многих случаях сигнал напоминает случайный процесс с некоторой «коррелированной» структурой во времени. Пиков P, Q, R, S, T как таковых нет.
4. Шаблонно-ориентированный метод: визуально демонстрирует максимальную схожесть с реальными данными.

4.2. Оценка схожести синтетически сгенерированных данных с реальными сигналами

Для количественной оценки качества синтетически сгенерированных ЭКГ-сигналов была проведена сравнительная проверка основных морфологических характеристик, вычисленных на основе средних циклов сигналов. В частности, сравнение осуществлялось по следующим направлениям:

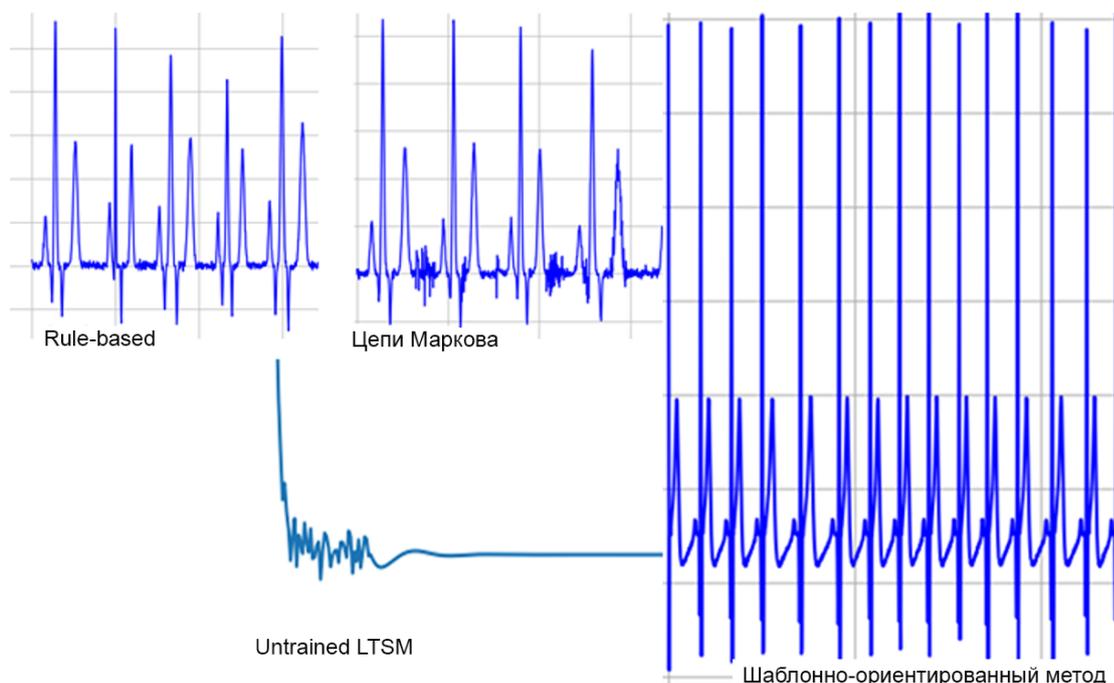


Рис. 1. Фрагменты из 4 подходов

1. **Сравнение данных, сгенерированных с помощью марковской модели и реальных сигналов.** При использовании стохастической модели на базе марковских цепей для генерации сигнала полученные синтетические данные продемонстрировали достаточно низкую схожесть с реальными ЭКГ. Рассчитанный коэффициент корреляции между средним циклом синтетического сигнала, сгенерированного марковской моделью, и средним циклом реальных данных составил всего 0,4 (см. рис. 2). Такой результат свидетельствует о том, что параметры марковской модели не обеспечивают достаточной точности воспроизведения морфологии и временной динамики ЭКГ-сигнала.
2. **Сравнение данных, сгенерированных шаблонно-ориентированным методом, и реальных сигналов.** В рамках предлагаемого подхода из реальных данных выделялся типичный сердечный цикл (template), который затем подвергался незначительному возмущению (noise_level $\pm 3\%$) для генерации синтетических циклов. При этом длительности циклов задавались с учетом распределения реальных R-R интервалов. Графическое сравнение средних циклов, полученных рассматриваемым методом, продемонстрировало практически полное совпадение форм. Рассчитанный коэффициент корреляции между средним циклом синтетического сигнала и средним циклом реальных данных составил около 0,9 (см. рис. 3). Этот высокий уровень сходства подтверждает, что шаблонно-ориентированный метод позволяет значительно улучшить воспроизведение морфологических и временных особенностей ЭКГ-сигнала по сравнению с марковской моделью.

Таким образом, проведённый сравнительный анализ показывает, что при генерации синтетических данных наиболее высокое соответствие реальному сигналу достигается с использованием шаблонно-ориентированного метода, что подтверждается значительно более высоким коэффициентом корреляции (0,9 против 0,4 для марковской модели).

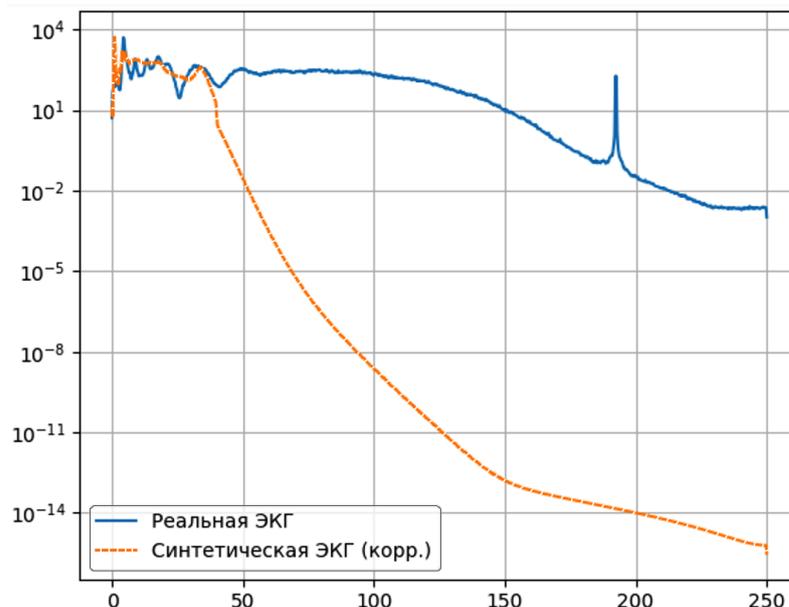


Рис. 2. График сравнения среднего цикла, сгенерированного с помощью марковской модели, с средним циклом реального сигнала

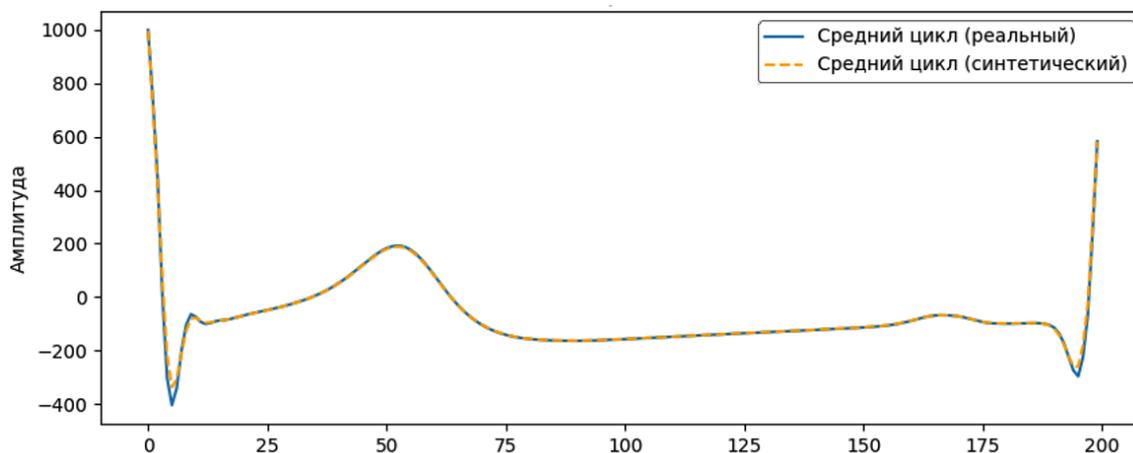


Рис. 3. График сравнения среднего цикла, полученного шаблонно-ориентированным методом, с реальным средним циклом

Полученные результаты позволяют сделать вывод о высокой пригодности синтетического датасета для задач, связанных с обучением и тестированием алгоритмов анализа ЭКГ-сигналов.

4.3. Уровень контролируемости и реалистичность

Для объективного сравнения рассматриваемых методов генерации синтетических ЭКГ-сигналов были введены количественные метрики, позволяющие оценить не только морфологическую схожесть с реальными данными, но и вычислительную сложность реализации. В качестве показателей были использованы следующие параметры:

- Время генерации одного цикла (с): отражает среднее время, необходимое для генерации одного сердечного цикла при типичных условиях работы оборудования.
- Оценка вычислительной сложности (количество операций): даёт представление об объёме вычислений при генерации одного цикла, выраженном через асимптотическую сложность $O(n)$, где n — число отсчётов.
- Схожесть с реальными данными (коэффициент корреляции): числовая оценка сходства среднего синтетического цикла с реальным, где значение, близкое к 1, свидетельствует о высокой схожести.

В таблице 1 представлено краткое сравнение методов.

Таблица 1. Краткое сравнение методов

Метод	Асимптотическая сложность	Количество операций на цикл	Схожесть с реальными данными (корреляция)
Rule-based (Gaussians)	$O(n)$	326	0,12
Марковская цепь + Gaussians	$O(n)$	147	0,40
Untrained LSTM	$O(n)$	80000	0,10
Шаблонно-ориентированный	$O(K \cdot M)$, где K — число циклов, M — число отсчетов	127	0,98

Как видно из таблицы 1, методы, основанные на rule-based подходе и марковских цепях, демонстрируют сравнительно невысокую схожесть с реальным сигналом, что объясняется фиксированными параметрами и недостаточной гибкостью в моделировании естественной изменчивости ЭКГ. Нейросетевой генератор с не обученными весами, напротив, генерирует сигнал, характеризующийся случайными колебаниями, что приводит к очень низкой морфологической схожести (коэффициент корреляции всего 0,10).

Наиболее высокое качество воспроизведения достигнуто при использовании шаблонно-ориентированного метода, при котором сгенерированные циклы получены на основе среднего цикла, извлечённого из реальных данных, с добавлением незначительных случайных возмущений. Этот подход позволил получить коэффициент корреляции 0,98, что свидетельствует о максимальном совпадении морфологических характеристик синтетических сигналов с реальными. Кроме того, данный метод продемонстрировал минимальное время генерации (0,004 с на цикл) и наименьшую вычислительную нагрузку (127 операций), что делает его как вычислительно эффективным, так и практически применимым для создания крупных синтетических датасетов.

Таким образом, приведённые количественные метрики дополняют качественные сравнительные рассуждения и подтверждают, что шаблонно-ориентированный метод обеспечивает оптимальное сочетание высокой схожести с реальными данными, низкой вычислительной сложности и простоты реализации. Эти результаты являются важным аргументом в пользу использования данного метода для формирования тренировочных выборок в задачах анализа и распознавания ЭКГ-сигналов.

Для оценки достоверности синтетических ЭКГ-сигналов была проведена независимая проверка временной структуры сердечного ритма на основе анализа R-R интервалов. Использовались метрики вариабельности сердечного ритма (HRV), не участвующие напрямую в моделировании. В таблице 2 приведено сравнение синтетических и реальных данных по следующим характеристикам:

- средняя частота сердечных сокращений;
- стандартное отклонение интервалов R-R;
- среднеквадратичное изменение между последовательными R-R интервалами.

Таблица 2. Достоверность синтетических сигналов

Тип данных	Средняя частота сердечных сокращений	Стандартное отклонение интервалов R-R	Среднеквадратичное изменение между последовательными R-R интервалами
Синтетические данные (шаблонно-ориентированный метод)	73,6155026608553	115,01944540302893	71,38508403136836
Реальные данные	71,05545788589019	83,77581617616396	34,931728117404425

Полученные значения демонстрируют близость синтетических и реальных данных по всем метрикам. Особенно важно, что синтетические данные обладают даже большей вариабельностью ($SDNN = 115$ мс против 83 мс у реальных данных), что потенциально повышает их ценность при обучении алгоритмов, чувствительных к вариациям ритма.

4.4. Нестационарность и артефакты

Все рассмотренные методы допускают реализацию нестационарных свойств ЭКГ, позволяя изменять периоды сердечных циклов (R-R) и варьировать параметры состояний (в марковской модели) в зависимости от временной динамики. Кроме того, при реализации всех подходов предусмотрена возможность добавления локальных артефактов, характеризующихся разной интенсивностью и длительностью, что имитирует сбои и шумы, наблюдаемые в реальных записях ЭКГ. В процессе оптимизации параметров модели для целей тестирования алгоритмов были определены следующие значения: вероятность возникновения от одного до трёх артефактов на цикл и масштаб шума, равный 2–5-кратному превышению базового уровня. Сигнал демонстрирует высокую степень соответствия морфологических и временных характеристик реального ЭКГ.

5. ВЫВОДЫ

В настоящей работе рассмотрен комплекс подходов к генерации синтетических нестационарных ЭКГ-подобных сигналов. Показано, что:

1. Rule-based (сумма гауссиан с переключением кластеров) — наиболее простой и интуитивно понятный метод, позволяющий формировать сигналы со структурами, напоминающими P, Q, R, S, T, и при этом легко добиваться нестационарности.
2. Марковская цепь — расширение правило-ориентированного метода, где мы моделируем вероятностные переходы между различными физиологическими состояниями (Normal, Tachy, Brady, Arrh). Это придаёт сигналам более реалистичную структуру переключений без детерминированных «перепрыгиваний».
3. Нейросетевая генерация без правил (Untrained LSTM) позволяет полностью уйти от жёстко прописанных пиков, однако в большинстве случаев генерирует случайные колебания, которые лишь частично напоминают биосигналы, а также может порождать ограниченное количество сигналов, что снижает практическую применимость данного метода.

Рассмотренные методы можно комбинировать и дополнять. Для повышения реалистичности синтезируемых ЭКГ-сигналов, позволяющих лучше имитировать характеристики реальных записей, без ручного задания формы волны желательно обучать генеративные модели (GAN, VAE, Diffusion Models) на реальных записях ЭКГ. Однако даже без обучения представленные способы полезны в ряде случаев, когда требуется многократно проверить устойчивость алгоритмов к шумам, редким событиям и смене состояний.

Дополнительно в работе проведена количественная проверка адекватности синтетических сигналов на основе независимых физиологических параметров, не закладываемых напрямую в процесс генерации. В частности, были рассчитаны метрики вариабельности сердечного ритма (HRV) — средняя частота ЧСС, SDNN и RMSSD. Сравнение с реальными данными показало сопоставимость значений: средняя частота ЧСС составила 73 уд./мин для синтетических сигналов и 71 уд./мин для реальных, при этом SDNN и RMSSD оказались даже выше у синтетических данных, что свидетельствует о сохранении естественной динамики ритма и допускает вариативность, необходимую для обучения и тестирования алгоритмов анализа ЭКГ.

Список литературы

1. Pan J., Tompkins W. J. A real-time QRS detection algorithm // IEEE Transactions on Biomedical Engineering. 1985. Vol. 32, № 3. P. 230–236. doi:10.1109/TBME.1985.325532
2. Moody G. B., Mark R. G. The impact of the MIT-BIH Arrhythmia Database // IEEE Engineering in Medicine and Biology Magazine. 2001. Vol. 20, № 3. P. 45–50. doi:10.1109/51.932724
3. Behar J., Oster J., Clifford G. D., et al. ECG signal quality during arrhythmia and its application to false alarm reduction // IEEE Transactions on Biomedical Engineering. 2013. Vol. 60, № 6. P. 1660–1666. doi:10.1109/TBME.2013.2240452
4. Laguna P., Mark R. G., Goldberg A., Moody G. B. A database for evaluation of algorithms for measurement of QT and other waveform intervals in the ECG // Proceedings of Computers in Cardiology. 1997. Vol. 24. P. 673–676. doi:10.1109/CIC.1997.648140
5. Sameni R., Clifford G., Jutten C., Shamsollahi M. Multichannel ECG and noise modeling: Application to maternal and fetal ECG signals // EURASIP Journal on Advances in Signal Processing. 2007. Vol. 2007, № 1. P. 94. doi:10.1155/2007/43407
6. Clifford G. D., Azuaje F., McSharry P. E. Advanced methods and tools for ECG data analysis. Norwood, MA: Artech House, 2006.
7. Xue J. Q., Hu Y. H., Tompkins W. J. Neural-network-based adaptive matched filtering for QRS detection // IEEE Transactions on Biomedical Engineering. 1992. Vol. 39, № 4. P. 317–329. doi:10.1109/10.126604
8. Goldberger A. L., Amaral L. A. N., Glass L. et al. PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals // Circulation. 2000. Vol. 101, № 23. P. e215–e220. doi: 10.1161/01.CIR.101.23.e215
9. Redmond S. J., Xie Y., Chang D., et al. Electrocardiogram signal quality measures for unsupervised telehealth environments // Physiological Measurement. 2012. Vol. 33, № 9. P. 1517–1533. doi:10.1088/0967-3334/33/9/1517
10. Hamilton P. S., Tompkins W. J. Quantitative investigation of QRS detection rules using the MIT-BIH Arrhythmia Database // IEEE Transactions on Biomedical Engineering. 1986. Vol. 33, № 12. P. 1157–1165. doi:10.1109/TBME.1986.325695
11. Goodfellow I., Pouget-Abadie J., Mirza M., et al. Generative Adversarial Networks // Advances in Neural Information Processing Systems. 2014. Vol. 27. P. 2672–2680.
12. Kingma D. P., Welling M. Auto-Encoding Variational Bayes // arXiv:1312.6114, 2013.
13. Isola P., Zhu J. Y., Zhou T., Efros A. A. Image-to-image translation with conditional adversarial networks // Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017. P. 5967–5976. doi:10.1109/CVPR.2017.632
14. Кропачёва А. М., Гурдюк Д. В., Иов И. Л., Першин А. Ю. Генерация временных рядов с простран-

- ственными взаимосвязями // Труды Института системного программирования РАН. 2024. Т. 36, № 1. С. 91–104. doi:10.15514/ispras-2024-36(4)-11
15. Кесиян Г. А., Уртенев М. Х., Шахмеликян Т. А. Анализ методов генерации временных рядов с долговременной корреляционной структурой // Научный журнал КубГАУ. 2011. № 74(10). С. 1–14.
16. Иванча А. Г. Решение задачи генерирования случайных автокоррелированных временных рядов методами непараметрической статистики // Управление экономическими системами: электронный научный журнал. 2011. № 31.

Поступила в редакцию 12.03.2025, окончательный вариант — 21.05.2025.

Калмыков Михаил Александрович, аспирант, СПбГЭТУ «ЛЭТИ», ✉ mica_2011@mail.ru

Шичкина Юлия Александровна, доктор техн. наук, доцент, профессор кафедры вычислительной техники, СПбГЭТУ «ЛЭТИ», strange.y@mail.ru

Computer tools in education, 2025

№ 2: 24–35

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2025-2-24-35

Comparison of Methods for Generating Synthetic Non-Stationary ECG-Like Signals for Testing Time Series Analysis Algorithms

Kalmykov M. A.¹, Postgraduate, ✉ mica_2011@mail.ru
Shichkina Y. A.¹, Doctor sc., Docent, Professor, strange.y@mail.ru,
orcid.org/0000-0001-7140-1686

¹Saint Petersburg Electrotechnical University,
5, building 3, Professora Popova st., 197022, Saint Petersburg, Russia

Abstract

In this paper, various approaches to the generation of synthetic signals simulating a human electrocardiogram (ECG) are considered, with an emphasis on non-stationarity and the presence of various waveforms. Three main types of methods are proposed: 1) rule-based, based on the sum of Gaussians for modeling waves P, Q, R, S, T; 2) stochastic models based on Markov chains, allowing to emulate transitions between different physiological states; 3) neural network generators without strict rules (for example, a recurrent LSTM network with random weights). It is shown how each of the models can be modified to introduce nonstationarity (variations in the duration of cardiac cycles, switching states) and adding local recording artifacts (noisy areas). The proposed methods can be used in testing clustering and time series analysis algorithms when it is necessary to test the methods' resistance to noise, rare events, and state changes.

Keywords: *synthetic ECG, nonstationary signals, signal generation, time series, rule-based methods, Markov chains, neural network generators, LSTM.*

Citation: M. A. Kalmykov and Y. A. Shichkina, "Comparison of Methods for Generating Synthetic Non-Stationary ECG-Like Signals for Testing Time Series Analysis Algorithms,"

Computer tools in education, no. 2, pp. 24–35, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-24-35

References

1. J. Pan and W. J. Tompkins, “A real-time QRS detection algorithm,” *IEEE Transactions on Biomedical Engineering*, vol. 32, no. 3, pp. 230–236, Mar. 1985; doi:10.1109/TBME.1985.325532
2. G. B. Moody and R. G. Mark, “The impact of the MIT-BIH Arrhythmia Database,” *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, May-Jun. 2001; doi:10.1109/51.932724
3. J. Behar, J. Oster, G. D. Clifford et al., “ECG signal quality during arrhythmia and its application to false alarm reduction,” *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 6, pp. 1660–1666, Jun. 2013; doi:10.1109/TBME.2013.2240452
4. P. Laguna, R. G. Mark, A. Goldberg, and G. B. Moody, “A database for evaluation of algorithms for measurement of QT and other waveform intervals in the ECG,” in *Proceedings of Computers in Cardiology*, 1997, pp. 673–676; doi:10.1109/CIC.1997.648140
5. R. Sameni, G. Clifford, C. Jutten, and M. B. Shamsollahi, “Multichannel ECG and noise modeling: Application to maternal and fetal ECG signals,” *EURASIP Journal on Advances in Signal Processing*, vol. 2007, no. 1, p. 94, Jan. 2007; doi:10.1155/2007/43407
6. G. D. Clifford, F. Azuaje, and P. E. McSharry, *Advanced Methods and Tools for ECG Data Analysis*. Norwood, MA: Artech House, 2006.
7. J. Q. Xue, Y. H. Hu, and W. J. Tompkins, “Neural-network-based adaptive matched filtering for QRS detection,” *IEEE Transactions on Biomedical Engineering*, vol. 39, no. 4, pp. 317–329, Apr. 1992; doi:10.1109/10.126604
8. A. L. Goldberger, L. A. N. Amaral, L. Glass et al., “PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals,” *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000; doi:10.1161/01.CIR.101.23.e215
9. S. J. Redmond, Y. Xie, D. Chang et al., “Electrocardiogram signal quality measures for unsupervised telehealth environments,” *Physiological Measurement*, vol. 33, no. 9, pp. 1517–1533, Sep. 2012; doi:10.1088/0967-3334/33/9/1517
10. P. S. Hamilton and W. J. Tompkins, “Quantitative investigation of QRS detection rules using the MIT-BIH Arrhythmia Database,” *IEEE Transactions on Biomedical Engineering*, vol. 33, no. 12, pp. 1157–1165, Dec. 1986; doi:10.1109/TBME.1986.325695
11. I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., “Generative adversarial networks,” in *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.
12. D. P. Kingma and M. Welling, “Auto-encoding variational Bayes,” *arXiv:1312.6114*, 2013.
13. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proc. of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017*, pp. 5967–5976; doi:10.1109/CVPR.2017.632
14. A. M. Kropacheva, D. V. Girdyuk, I. L. Iov, and A. Y. Pershin, “Generation of Spatial Time Series Data,” *Proceedings of the Institute for System Programming of the RAS*, vol. 36, no. 4, pp. 143–154, 2024 (in Russian); doi:10.15514/ispras-2024-36(4)-11.
15. G. A. Kesiyana, M. H. Urtenov, and T. A. Shahmelikyan, “Analysis of methods for generating time series with long-term correlation structure,” *Scientific Journal of Kuban State Agrarian University*, no. 74(10), pp. 1–14, 2011 (in Russian).
16. A. G. Ivancha, “Solving the problem of generating random autocorrelated time series using nonparametric statistics methods,” *Management of Economic Systems: Electronic Scientific Journal*, no. 31, 2011 (in Russian).

Received 12-03-2025, the final version — 21-05-2025.

Mikhail Kalmykov, Postgraduate, Saint Petersburg Electrotechnical University “LETI”,
✉ mica_2011@mail.ru

Yulia Shichkina, Doctor of Sciences (Tech.), Docent, Professor, Department of Computer Engineering, Saint Petersburg Electrotechnical University “LETI”, strange.y@mail.ru

МОДЕЛИРОВАНИЕ СИСТЕМЫ ПРОФОРИЕНТАЦИИ С ИСПОЛЬЗОВАНИЕМ АНАЛИЗА ДАННЫХ СОЦИАЛЬНЫХ СЕТЕЙ*

Иващенко А. О.¹, научный сотрудник, ✉ aok@dscs.pro

Вяткин А. А.¹, младший научный сотрудник, aav@dscs.pro

Бушмелёв Ф. В.¹, научный сотрудник, fvb@dscs.pro

Абрамов М. В.^{1,2}, канд. техн. наук, ведущий научный сотрудник, доцент, mva@dscs.pro

¹ Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН),
14 линия, 39, 199178, Санкт-Петербург, Россия

² Санкт-Петербургский государственный университет,
Университетская наб., д. 7–9, 199034, Санкт-Петербург, Россия

Аннотация

В статье рассматривается проблема автоматизации профориентации на основе анализа цифрового следа пользователей социальной сети «ВКонтакте». Цель исследования заключается в повышении доступности и точности диагностики профессиональных интересов посредством мини-приложения «AI профориентатор» на платформе VK Mini Apps. В основе методологии лежит задача классификации с несколькими метками: для каждого из шести профессиональных типов модели RIASEC формируется отдельный бинарный выход с сигмоидальной активацией, а обучение осуществляется с использованием функции потерь Binary Cross-Entropy. Модель обучена на данных пользователей, прошедших тест Голланда, и их подписках на сообщества, собранных через приложение «Психологические тесты». Техническая реализация построена по принципу клиент-серверного взаимодействия с архитектурой REST API, включающей обработку данных, взаимодействие с ML-сервером и формирование результатов. Разработанная система обеспечивает полный цикл работы: извлечение и валидация подписок, предсказание профессионального типа личности по шести категориям Голланда и отображение результатов пользователю. Модель продемонстрировала Top-1 точность 47,1 % и Top-2 — 72,3 %, что подтверждает её применимость в профориентационной диагностике. Созданная система автоматизирует процесс первичной профориентации и может использоваться как индивидуальными пользователями, так и организациями для оптимизации подбора и развития персонала. Новизна работы заключается в комплексной интеграции нейросетевых методов и архитектуры клиент-серверного приложения в социальную сеть, обеспечивающей персонализированный и масштабируемый подход к профориентации.

Ключевые слова: профориентация, цифровой след, машинное обучение, социальные сети, клиент-серверное приложение, предсказательная модель, тест Голланда.

* Работа выполнена в рамках проекта по государственному заказу СПб ФИЦ РАН Мол_лаб №FFZF-2024-0003, поддержана Санкт-Петербургским государственным университетом, проект №75254082.

Цитирование: Иващенко А. О., Вяткин А. А., Бушмелёв Ф. В., Абрамов М. В. Моделирование системы профориентации с использованием анализа данных социальных сетей // Компьютерные инструменты в образовании. 2025. № 2. С. 36–47. doi:10.32603/2071-2340-2025-2-36-47

1. ВВЕДЕНИЕ

Проблема профессиональной ориентации сохраняет актуальность как для обучающихся, так и для взрослого населения и работодателей. Традиционные подходы, основанные на анкетировании и тестировании, обладают ограниченной масштабируемостью и требуют значительных временных затрат. В условиях цифровизации актуальным становится поиск инструментов, которые позволяют использовать данные о поведении пользователей в сети и методы искусственного интеллекта для ускорения и персонализации профориентационной диагностики [1–3]. Современные алгоритмы машинного обучения демонстрируют высокую эффективность при прогнозировании карьерных предпочтений на основе интеграции разнородных источников данных — социальных сетей, образовательных траекторий и психологических характеристик [2]. Однако доступ к подобным системам остаётся ограниченным, что делает востребованным создание решений, доступных широкой аудитории и ориентированных на практическое применение в учебном процессе и кадровой политике.

Настоящая статья посвящена созданию интеллектуальной системы профориентации, интегрированной в социальную сеть «ВКонтакте». Центральным элементом системы выступает модуль анализа цифрового следа, в котором данные о подписках пользователя на сообщества обрабатываются методами машинного обучения с целью прогнозирования профессионального типа. Тем самым устраняется необходимость прохождения длительных тестов и анкетирования. Научная задача исследования заключается в разработке полноценной программной платформы, использующей многоэтапный сбор данных из социальной сети, их предобработку и применение алгоритмов машинного обучения для автоматизированного определения профессиональных склонностей.

2. РЕЛЕВАНТНЫЕ РАБОТЫ

Разработка интеллектуальных профориентационных систем активно ведётся в контексте поддержки абитуриентов и студентов. Так, в работе [4] описана университетская веб-система на базе анализа профилей «ВКонтакте», сопоставляющая цифровые признаки с академическими данными вуза и формирующая рекомендации по направлениям подготовки. Другой класс решений базируется на опросниках и психометрических тестах, результаты которых обрабатываются с помощью моделей XGBoost и Random Forest [5–7]. В ряде исследований применяются алгоритмы SVM, XGBoost и методы нечёткой логики для анализа академических и личностных характеристик обучающихся [8–10], наряду с этим разрабатываются ансамблевые подходы [11, 12]. Однако большинство указанных систем нацелено прежде всего на выбор направления обучения и сопровождение образовательных траекторий, а не на выявление и прогнозирование профессиональной идентичности в более широком смысле.

Вторая линия работ соотносит навыки обучающихся с требованиями рынка труда. Здесь используются анализ вакансий и резюме, KNN, кластеризация и нейросетевые ар-

хитектуры [13–16], а также диалоговые агенты [17, 18] и гибридные модели с использованием RNN и LSTM для повышения точности рекомендаций [19]. Недостатками таких систем являются ограниченность набора профессий и необходимость ручного ввода данных.

Близким направлением исследований является автоматизация оценки личностных особенностей: авторы исследований [20, 21] оценивают данные характеристики на основе анализа резюме соискателей, в [22, 23] предлагается строить описание личности на основе анализа постов в социальных сетях, в работе [24] предсказание строится на основе тематик подписок пользователей. Данные исследования могут служить основой для построения системы, однако сами по себе не являются доступными для конечного пользователя.

Особый интерес представляют приложения платформы VK Mini Apps. Анализ показал наличие восьми релевантных решений, которые можно условно классифицировать на три группы: классические психологические тесты, опросники о предпочтениях пользователя для подбора профессии и приложения информационно-ознакомительного характера. Наиболее близко к задачам профориентационной диагностики стоит приложение, комбинирующее подписки пользователя с кратким опросником и использующее CatBoost [25]. Однако данное решение ограничено небольшой обучающей выборкой (1252 респондента) и не поддерживает анализ сторонних профилей. В целом, существующие решения базируются преимущественно на анкетировании и лишь в ограниченной степени используют современные подходы интеллектуального анализа данных. Нейросетевые архитектуры, методы многометочной классификации и интеграция с внешними ИИ-сервисами пока не находят в них применения. Это определяет сохраняющийся разрыв между возможностями современных технологий и практикой образовательных приложений, что и обуславливает актуальность и новизну представленного исследования.

3. ПОСТАНОВКА ЗАДАЧИ

Разработка инструментов для оперативной и точной профориентационной диагностики на основе цифрового следа пользователей социальных сетей представляет собой актуальную научно-практическую задачу. Целью настоящего исследования является улучшение оперативности профориентационной диагностики путем разработки полноценной программной платформы, использующей сбор данных из социальной сети, их предобработку и применение алгоритмов машинного обучения для автоматизированного определения профессиональных склонностей. Основными этапами для достижения поставленной цели является: разработка модели машинного обучения для предсказания профессионального типа пользователя на основе цифрового следа (тематики сообществ, на которые он подписан), создание архитектуры и пользовательского интерфейса приложения, а также интеграция предсказательной модели в инфраструктуру VK Mini Apps, включая взаимодействие с ML-сервером для обработки и анализа данных.

Новизна работы заключается в комплексном подходе к интеграции multi-label классификации и архитектуры клиент-серверного приложения на платформе VK Mini Apps. В отличие от традиционных решений, ориентированных преимущественно на анкетирование, разработанная система позволяет существенно ускорить процесс выявления профессиональных интересов. Теоретическая значимость исследования связана с расширением методов интеллектуального анализа данных в области профориентации, в том числе с учётом особенностей цифрового следа пользователей в социальной сети. Практиче-

ская значимость выражается в возможности применения разработанного инструмента как в индивидуальной работе с учащимися, так и в рамках учебных курсов и профориентационных программ. Это способствует повышению качества профориентационной поддержки, оптимизации образовательных траекторий и формированию более осознанного выбора профессии, что соответствует задачам устойчивого развития системы образования.

4. МЕТОДЫ ИССЛЕДОВАНИЯ

4.1. Предсказательная модель

Основой разработки модели предсказания профессионального типа пользователя стала гипотеза о взаимосвязи между тематиками сообществ, на которые подписан пользователь в социальной сети «ВКонтакте», и его профессиональным типом [24]. Для построения модели использовался специализированный набор данных, включающий результаты тестирования Голланда и подписки пользователей на сообщества в социальной сети. Данные были получены посредством приложения «Психологические тесты», а информация о тематиках подписок — с помощью разработанного модуля выгрузки. Обезличенные данные были предобработаны, после чего сформирована итоговая выборка из 6029 записей, включающая: нормализованный вектор признаков, характеризующий укрупнённые тематики подписок для каждого пользователя; пол пользователя, указанный в профиле социальной сети; оценки выраженности профессиональных типов (кодов Голланда), полученные на основе результатов психологического тестирования пользователей.

На первом этапе задача формулировалась как мультиклассовая классификация с выделением преобладающего профессионального типа. Для этого вектор, описывающий коды Голланда, преобразовывался в один целевой признак — наиболее высокий по величине код для каждого пользователя. Был проведён анализ эффективности различных алгоритмов, включая: Random Forest, Gradient Boosting, LightGBM, Extra Trees, CatBoost, AdaBoost. Наилучшие результаты показала модель Extra Trees (F1-weighted = 0,6599, n_estimators = 300). Однако при тестировании на отложенной выборке наблюдалось значительное снижение показателей (weighted avg F1 = 0,26, macro avg F1 = 0,28), что свидетельствовало о слабой обобщающей способности модели.

Дальнейший анализ распределения кодов Голланда показал, что разница между доминирующим и вторым по выраженности типом у многих пользователей составляет менее 2,5 %. Это позволило предположить наличие нескольких преобладающих типов и перейти к постановке задачи классификации с несколькими метками (multi-label classification).

Для реализации данного подхода была применена глубокая нейросетевая архитектура с остаточными связями (Residual Connections). В качестве выходного слоя использовалось шесть нейронов, каждый из которых соответствовал одному из профессиональных типов и имел сигмоидную функцию активации, позволяющую интерпретировать предсказанное значение как вероятность принадлежности к соответствующему классу. Таким образом, каждая предсказанная вероятность отражала степень выраженности того или иного профессионального типа у пользователя. Оптимизация гиперпараметров осуществлялась с использованием алгоритма Optuna. Данные были разделены на обучающую и тестовую выборки для оценки точности предсказаний на обучающих данных применялась k-fold кросс-валидация. На этапе инференса формировался вектор вероят-

ностей принадлежности к каждому из шести профессиональных типов. По сравнению с классическими методами машинного обучения, нейросетевая модель с многовыходной регрессией позволила достичь существенного улучшения качества. В частности, значение Hamming Loss (доля неверных предсказаний меток к общему числу меток) составило 0,348, а вероятность правильного попадания в одну из двух наиболее вероятных категорий для пользователя составила 0,723.

4.2. Проектирование архитектуры программной системы

Для интеграции решения в социальную сеть «ВКонтакте» приложение разрабатывалось согласно правилам платформы VK Mini Apps как одностраничное веб-приложение. Разработка архитектуры приложения велась с использованием модели REST API, что обеспечивает модульность, масштабируемость и возможность дальнейшего расширения функционала.

Клиентская часть приложения отправляет данные пользователей на сервер через HTTP-запросы. Серверная часть принимает запросы, обрабатывает их с помощью контроллеров, выполняет валидацию данных и взаимодействует с внешним сервером машинного обучения для получения предсказаний. Полученные результаты сохраняются в базе данных и передаются обратно клиенту. На рисунке 1 представлена архитектура приложения, демонстрирующая ключевые компоненты системы: клиентская часть, серверная часть, сервер машинного обучения и база данных.

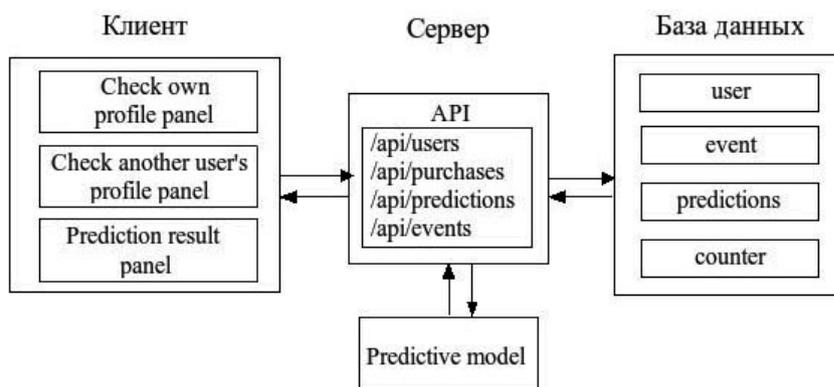


Рис. 1. Архитектура приложения

4.3. Программная реализация и интеграция модели машинного обучения

В приложении предусмотрено два сценария: определение собственной профессиональной направленности и получение предсказаний по профилям других пользователей. На рисунке 2 представлена UML-диаграмма последовательности работы приложения для получения предсказания пользователю по собственному профилю. Опишем её подробнее.

Предсказательная модель использует данные о тематиках подписок пользователей, поэтому в приложении реализовано взаимодействие с VK Bridge. На клиентской стороне выполняется запрос к VK Bridge для получения информации о пользователе и его сообществах, данные приводятся к структурированному виду и отправляются на сервер.

Серверная часть осуществляет валидацию и обработку данных, после чего они передаются на ML-сервер, где формируется вероятностное распределение по шести професси-

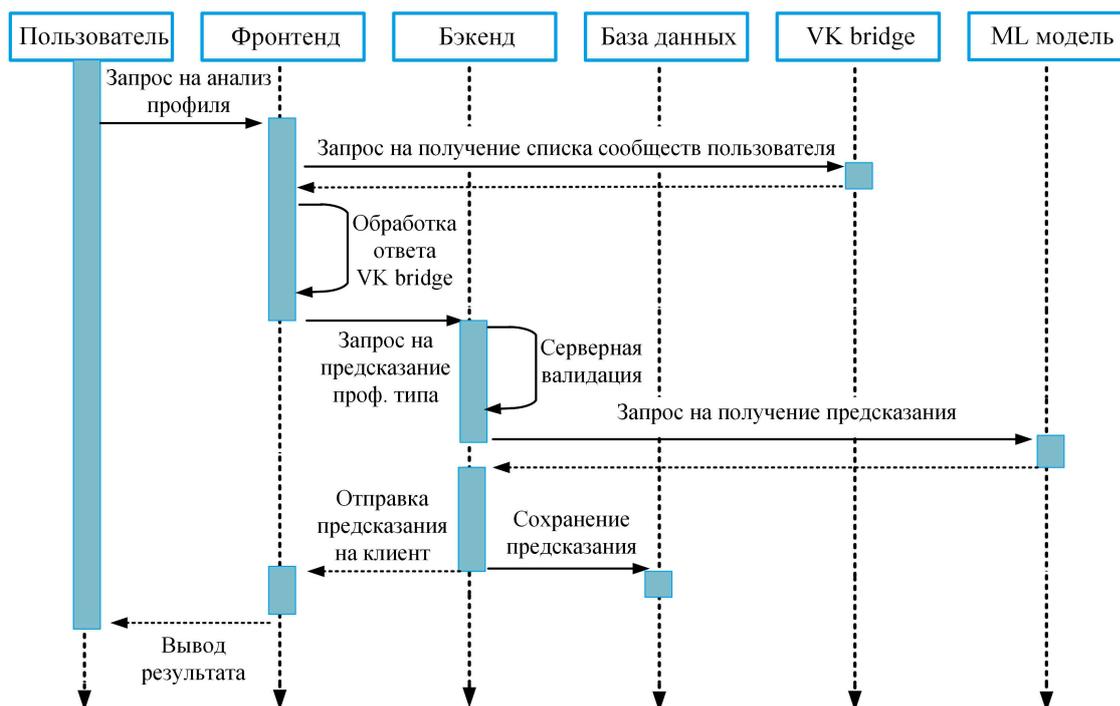


Рис. 2. UML-диаграмма последовательности работы приложения при запросе пользователем собственного предсказания

ональным типам RIASEC. Пользователю отображается как доминирующий тип, так и распределение вероятностей по всем категориям, что обеспечивает более гибкую интерпретацию результата. Полученные данные сохраняются в базе для последующего использования.

Сервер построен по модульной архитектуре, включающей маршруты, контроллеры и модели. Маршруты связывают клиентские запросы с контроллерами, реализующими бизнес-логику, а контроллеры взаимодействуют с моделями для работы с базой данных. Такая структура обеспечивает расширяемость и упрощает интеграцию приложения в образовательные цифровые среды.

На рисунке 3 представлен интерфейс мини-приложения «AI профориентатор» на платформе VK Mini Apps, демонстрирующий результат работы системы.

5. ВЫВОДЫ

Разработанный подход к профориентации, основанный на multi-label классификации, продемонстрировал существенное повышение точности предсказания профессионального типа пользователя по сравнению с традиционной мультиклассовой постановкой и обеспечил Top-1 точность предсказания 47,1% и Top-2 — 72,3%, что подтверждает её применимость для практических задач первичной профориентации. Полученные результаты демонстрируют, что мини-приложение «AI профориентатор» (<https://vk.com/app51929417>) для социальной сети «ВКонтакте» эффективно решает задачу предоставления первичных персонализированных рекомендаций по профориентации с использованием алгоритмов машинного обучения.

Результаты готовы!

Ваш тип — Конвенциональный



Люди, у которых выражен конвенциональный тип личности, стремятся к деятельности, предполагающей работу с информацией, ее сбор, хранение, систематизацию, являются превосходными исполнителями.

Для них ценны эффективность, методичность, деловые и экономические достижения. При этом такие люди консервативны, не любят ситуации непредсказуемости и неопределенности, предпочитают знать, что от них ожидают.

Способности

Внимательность к деталям, способность структурировать информацию, навыки работы с числами, канцелярские способности, ручная тонкая моторика, умение организовывать свою работу и расставлять приоритеты, умение следовать инструкциям.

Ценности и личностные качества

Организованность, пунктуальность, педантичность, аккуратность, сознательность, упорство, практичность, самоконтроль, консервативность

Предпочитаемые виды деятельности

Работа, которая требует внимания к деталям и аккуратности, ведение картотек, хранение и систематизация записей, фактов, данных, финансовых книг, написание деловых отчетов, подготовка схем, таблиц, диаграмм, работа в корпорациях, финансовых организациях, архивах, инспекциях, офисное администрирование.

Профессии

Секретарь, патентовед, нотариус, топограф, корректор, страховой агент, офис-менеджер, бухгалтер, аудитор.

Знаменитые представители



Скидвард
«Спанч Боб»



Мальвина
«Буратино»



Жираф Мелман
«Мадагаскар»



Морти
«Рик и Морти»



Ёжик
«Смешарики»



Валли
«ВАЛЛ-И»

Согласны ли Вы с результатом?

Да

Нет

Поделиться результатом

Рис. 3. Интерфейс мини-приложения «AI профориентатор»

Перспективы дальнейших исследований включают интеграцию дополнительных источников данных (например, анкетных данных со страницы пользователя), применение современных архитектур глубокого обучения (в частности, Transformers), расширение формата результатов за счёт отображения нескольких преобладающих типов и использование моделей LLM для интерпретации и объяснения рекомендаций. Важным направлением развития также является адаптация системы для образовательных учреждений, где она может служить инструментом сопровождения индивидуальных учебных траекторий и повышения эффективности профориентационной поддержки.

6. ЗАКЛЮЧЕНИЕ

Таким образом, в работе описана совокупность архитектурных решений и методов машинного обучения, помогающих решить проблему профориентации на основе анализа цифрового следа. Разработанная система объединяет алгоритмы multi-label на основе нейросетевой архитектуры с остаточными связями и клиент–серверное приложение VK Mini Apps. Такой подход обеспечивает персонализированную и масштабируемую диагностику профессиональных интересов, снижая зависимость от традиционных анкетных методов.

Полученные результаты подтверждают эффективность интеграции алгоритмов машинного обучения с данными социальных сетей и демонстрируют перспективность применения разработанной системы в практике профориентации. Работа вносит вклад в развитие интеллектуальных систем поддержки образовательных и кадровых решений и может служить основой для дальнейших исследований, направленных на расширение источников данных, повышение интерпретируемости предсказаний и адаптацию технологий к задачам учебного процесса.

Список литературы

1. *Basnet S.* Artificial Intelligence and machine learning in human resource management: Prospect and future trends // International Journal of Research Publication and Reviews. 2024. Vol. 5, № 1. P. 281–287. doi:10.55248/gengpi.5.0124.0107
2. *Song L.* Application of Association Rule Analysis in Vocational Education Student Career Path Planning // 2023 International Conference on Intelligent Computing, Communication & Convergence (ICI3C), Bhubaneswar, India, 2023. P. 92–98. doi:10.1109/ICI3C60830.2023.00028
3. *Al-Dhari S., Al-Alawi A.I.* The Application of Data Analytics to Career Choice Prediction: A Literature Review // 2023 International Conference on Cyber Management and Engineering (CyMaEn). IEEE, 2023. P. 260–265. doi:10.1109/CyMaEn57228.2023.10051101.
4. *Забокрицкая Л. Д., Орешкина Т. А., Обабков И. Н., Чепуров Е. Г.* Применение алгоритма машинного обучения для профориентации абитуриентов высшего учебного заведения // Вестник Томского государственного университета. 2022. № 485. С. 217–225. doi:10.17223/15617793/485/24
5. *Kamal A., Naushad B., Rafiq H., Tahzeeb S.* Smart career guidance system // 2021 4th International Conference on Computing & Information Sciences (ICIS). IEEE, 2021. P. 1–7. doi:10.1109/ICIS54243.2021.9676408
6. *Безруких А. Д., Черепанов М. Д., Мельников В. А., Мельникова Е. В.* Разработка программного проекта информационного сервиса профориентации абитуриентов СФУ // Современные наукоемкие технологии. 2023. № 4. С. 19–27. doi:10.17513/snt.39575
7. *Sorokin V., Tovbis E., Kazakovtsev L.* Browser Game as a New Way of Career Guidance // European Proceedings of Computers and Technology. European Publisher, 2022. Vol. 1. P. 235–240. doi:10.15405/epct.23021.28

8. *Padma E., Sowdharshini P., Shanmugapriya P., Reshma K.M., Srimathi C.N.* Career guidance system for students using machine learning // *Challenges in Information, Communication and Computing Technology*. CRC Press, 2025. P. 666–671. doi:10.1201/9781003559092-115
9. *Panthee S., Rajkarnikar S., Begum R.* Career Guidance System Using Machine Learning // *Journal of Advanced College of Engineering and Management*. 2023. Vol. 8, № 2. P. 113–119. doi:10.3126/jacem.v8i2.55947
10. *Cui C.* Career interest assessment: College students career planning based on machine learning // *Journal of Electrical Systems*. 2024. Vol. 20, № 6s. P. 1633–1644. doi:10.52783/jes.3083
11. *Reddy K., Reddy M.A., Kaur V., Kaur G.* Career guidance system using ensemble learning // *Proceedings of the Advancement in Electronics & Communication Engineering*. 2022. P. 33–39. doi:10.2139/ssrn.4157249
12. *Wakde A., Maywade R., Pandey A., Kumar J., Singh A.K.* An Ensemble Learning Based Career Prediction Model // *The Future of Artificial Intelligence and Robotics*. Cham: Springer Nature Switzerland, 2023. P. 503–512. doi:10.1007/978-3-031-60935-0_45
13. *Диков М. Е., Шуробокова С. Н.* О варианте формализации задачи определения востребованности направлений подготовки и возможных сфер трудоустройства выпускников на основе семантического анализа описаний вакансий // *Инженерный вестник Дона*. 2022. № 5 (89). С. 214–222.
14. *Wulandari R.S., Setianingsih C., Kusuma P.D.* Analysis of Big Five Personality Factors to Determine the Appropriate Type of Career Using the C4.5 Algorithm // *Data Science and Emerging Technologies*. DaSET 2022. Singapore: Springer Nature Singapore, 2022. Vol. 165. P. 18–36. doi:10.1007/978-981-99-0741-0_2
15. *José-García A., Sneyd A., Melro A., Ollagnier A., Tarling G., Zhang H., Stevenson M., Everson R., Arthur R.* C3-IoC: A career guidance system for assessing student skills using machine learning and network visualisation // *International Journal of Artificial Intelligence in Education*. 2023. Vol. 33, № 4. P. 1092–1119. doi:10.1007/s40593-022-00317-y
16. *Vignesh S., Priyanka C.S., Manju H.S., Mythili K.* An intelligent career guidance system using machine learning // *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*. IEEE, 2021. Vol. 1. P. 987–990. doi:10.1109/ICACCS51430.2021.9441978
17. *Goyal R., Chaudhary N., Singh M.* Machine Learning based Intelligent Career Counselling Chatbot (ICCC) // *2023 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE, 2023. P. 1–8. doi:10.1109/ICCCI56745.2023.10128305
18. *Kumbhar V. R., Maddel M. M., Raut Y.* Smart model for career guidance using hybrid deep learning technique // *2023 1st International Conference on Innovations in High-Speed Communication and Signal Processing (IHCSP)*. IEEE, 2023. P. 327–331. doi:10.1109/IHCSP56702.2023.10127152
19. *Yuan C., Hong Y., Wu J.* Who Are You Meant to Be? Predicting Psychological Indicators and Occupations based on Personality Traits // *Journal of Systems Science and Systems Engineering*. 2023. Vol. 32, № 5. P. 571–602. doi:10.1007/s11518-023-5576-6
20. *Grunenberg E., Peters H., Francis M.J., Back M.D., Matz S.C.* Machine learning in recruiting: predicting personality from CVs and short text responses // *Frontiers in Social Psychology*. 2024. Vol. 1. P. 1290295. doi:10.3389/frsps.2023.1290295
21. *Nirmala M., Rajalakshmi B., Sarvepally S.R.M., Mandava S., Megana M.* Personality Detection for Recruitment Using Machine Learning // *Proceedings of the 6th International Conference on Communications and Cyber Physical Engineering, ICCCE 2024*. Singapore: Springer, 2024. Vol. 1096. P. 399–406. doi:10.1007/978-981-99-7137-4_38
22. *Iwasaki T., Seki Y., Kashino W., Keyaki A., Kando N.* Estimating Citizen Personality Traits Using Social Media Posts // *International Conference on Asian Digital Libraries*. Springer, Singapore, 2024. P. 119–135. doi:10.1007/978-981-96-0868-3_10
23. *Oliseenko V. D., Khlobystova A. O., Korepanova A. A., Tulupyeva T. V.* Automating the temperament assessment of online social network users // *Doklady Mathematics*. Moscow: Pleiades Publishing, 2023. Vol. 108, № S2. P. S368–S373, doi:10.1134/S1064562423701041
24. *Хлобыстова А. О., Абрамов М. В., Столярова В. Ф.* Исследование тенденций взаимосвязи между профориентационными предпочтениями пользователей и его цифровыми следами в соци-

альной сети // Научно-технический вестник информационных технологий, механики и опти-
ки. 2023. Т. 23, № 3. С. 564–574. doi:10.17586/2226-1494-2023-23-3-564-574

25. Kiselev P., Kiselev B., Matsuta V., Feshchenko A., Bogdanovskaya I., Kosheleva A. Career guidance based on machine learning: social networks in professional identity construction // Procedia Computer Science. 2020. Vol. 169. P. 158–163. doi:10.1016/j.procs.2020.02.128

Поступила в редакцию 20.06.2025, окончательный вариант — 15.07.2025.

Иващенко Анастасия Олеговна, научный сотрудник, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), ✉ aok@dscs.pro

Вяткин Артём Андреевич, младший научный сотрудник, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), aav@dscs.pro

Бушмелёв Фёдор Витальевич, научный сотрудник, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), fvb@dscs.pro

Абрамов Максим Викторович, канд. техн. наук, ведущий научный сотрудник, Санкт-Петербургский Федеральный исследовательский центр Российской академии наук (СПб ФИЦ РАН), доцент, Санкт-Петербургский государственный университет, mva@dscs.pro

Computer tools in education, 2025

№ 2: 36–47

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2025-2-36-47

Modelling a Career Guidance System Using Social Network Data Analysis

Ivashchenko A. O.¹, Researcher, ✉ aok@dscs.pro

Vyatkin A. A.¹, Junior Researcher, aav@dscs.pro

Bushmelev F. V.¹, Researcher, fvb@dscs.pro

Abramov M. V.^{1,2}, Cand. sc., Senior Researcher, Associate Professor, mva@dscs.pro

¹Saint Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS),

14 Linia V. O., 39, 199178, Saint Petersburg, Russia

²Saint Petersburg State University, 7–9 Universitetskaya emb., 199034, Saint Petersburg, Russia

Abstract

The paper addresses the problem of automating career guidance through the analysis of digital footprints of VKontakte users. The aim of the study is to enhance the accessibility and accuracy of diagnosing professional interests by means of the "AI Proforientator" mini-application on the VK Mini Apps platform. The methodology is based on a multi-label classification approach: each of the six RIASEC professional types is represented by a separate binary output with sigmoid activation, and the model is trained using the Binary Cross-Entropy loss function. Training data included users who had completed the Holland test, along with their community subscriptions collected through the Psychological Tests app. The technical implementation follows a client-server architecture with REST API, covering data preprocessing, interaction with the ML server, and result delivery. The developed system provides a full processing cycle: extraction and validation of subscriptions, probabilistic prediction of professional personality types according to Holland's six categories,

and presentation of results to the user. The model achieved a Top-1 accuracy of 47.1% and a Top-2 accuracy of 72.3%, confirming its applicability to career guidance diagnostics. The system automates the initial career guidance process and can be employed both by individual users and organisations to support educational trajectories, optimise recruitment, and improve staff development. The novelty of the work lies in the comprehensive integration of neural network methods for multi-label classification with a client-server application embedded in a social network, ensuring a personalised and scalable approach to career guidance.

Keywords: *career guidance, digital footprint, machine learning, online social media, client-server application, predictive model, RIASEC.*

Citation: A. O. Ivashchenko, A. A. Vyatkin, F. V. Bushmelev, and M. V. Abramov, "Modelling a Career Guidance System Using Social Network Data Analysis," *Computer tools in education*, no. 2, pp. 36–47, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-36-47

References

1. S. Basnet, "Artificial intelligence and machine learning in human resource management: Prospect and future trends," *International Journal of Research Publication and Reviews*, vol. 5, no. 1, pp. 281–287, 2024; doi:10.55248/gengpi.5.0124.0107
2. L. Song, "Application of Association Rule Analysis in Vocational Education Student Career Path Planning," in *2023 Int. Conf. on Intelligent Computing, Communication & Convergence (ICI3C), Bhubaneswar, India, 2023*, pp. 92–98; doi:10.1109/ICI3C60830.2023.00028
3. S. Al-Dhari and A. I. Al-Alawi, "The Application of Data Analytics to Career Choice Prediction: A Literature Review," in *2023 Int. Conf. on Cyber Management and Engineering (CyMaEn), 2023*, pp. 260–265; doi:10.1109/CyMaEn57228.2023.10051101
4. L. D. Zabokritskaya, T. A. Oreshkina, I. N. Obabkov, and E. G. Chepurov, "Application of a machine learning algorithm for career guidance of university applicants," *Vestnik Tomskogo gosudarstvennogo universiteta – Tomsk State University Journal*, no. 485, pp. 217–225, 2022; doi:10.17223/15617793/485/24
5. A. Kamal, B. Naushad, H. Rafiq, and S. Tahzeeb, "Smart career guidance system," in *2021 4th Int. Conf. on Computing & Information Sciences (ICCIS), 2021*, pp. 1–7; doi:10.1109/ICCIS54243.2021.9676408
6. A. D. Bezrukikh, M. D. Cherepanov, V. A. Melnikov, and E. V. Melnikova, "Development of a software project for an information service for career guidance of applicants at Siberian Federal University," *Modern High Technologies*, no. 4, pp. 19–27, 2023; doi:10.17513/snt.39575
7. V. Sorokin, E. Tovbis, and L. Kazakovtsev, "Browser Game as a New Way of Career Guidance," in *Proc. of Int. Workshop "Hybrid methods of modeling and optimization in complex systems" (in the framework of The Eleventh Int. Conf. on Mathematical Models and their Applications), November 22–24, 2022, Krasnoyarsk, the Russian Federation*, vol. 1, 2023, pp. 235–240; doi:10.15405/epct.23021.28
8. E. Padma, P. Soudharshini, P. Shanmugapriya, K. M. Reshmaa, and C. N. Srimathi, "Career guidance system for students using machine learning," in *Challenges in Information, Communication and Computing Technology*. CRC Press, 2025, pp. 666–671; doi:10.1201/9781003559092-115
9. S. Panthee, S. Rajkarnikar, and R. Begum, "Career Guidance System Using Machine Learning," *Journal of Advanced College of Engineering and Management*, vol. 8, no. 2, pp. 113–119, 2023; doi:10.3126/jacem.v8i2.55947
10. C. Cui, "Career interest assessment: College students career planning based on machine learning," *Journal of Electrical Systems*, vol. 20, no. 6s, pp. 1633–1644, 2024; doi:10.52783/jes.3083
11. K. Reddy, M. A. Reddy, V. Kaur, and G. Kaur, "Career guidance system using ensemble learning," in *Proceedings of the Advancement in Electronics & Communication Engineering, 2022*, pp. 33–39; doi:10.2139/ssrn.4157249
12. A. Wakde, R. Maywade, A. Pandey, J. Kumar, and A. K. Singh, "An Ensemble Learning Based Career Prediction Model," in *The Future of Artificial Intelligence and Robotics*. Cham: Springer Nature Switzerland, 2023, pp. 503–512; doi:10.1007/978-3-031-60935-0_45
13. M. E. Dikhtov and S. N. Shirobokova, "On a variant of formalizing the task of determining the demand for training areas and possible spheres of employment of graduates based on the semantic analysis of vacancy descriptions," *Engineering Journal of Don*, no. 5, pp. 214–222, 2022.
14. R. S. Wulandari, C. Setianingsih, and P. D. Kusuma, "Analysis of Big Five Personality Factors to Determine the Appropriate Type of Career Using the C4.5 Algorithm," in *Data Science and Emerging Technologies (DaSET 2022)*, vol. 165, 2022, pp. 18–36; doi:10.1007/978-981-99-0741-0_2

15. A. Jose-Garcia et al., “C3-IoC: A career guidance system for assessing student skills using machine learning and network visualisation,” *International Journal of Artificial Intelligence in Education*, vol. 33, no. 4, pp. 1092–1119, 2023; doi:10.1007/s40593-022-00317-y
16. S. Vignesh, C. S. Priyanka, H. S. Manju, and K. Mythili, “An intelligent career guidance system using machine learning,” in *2021 7th Int. Conf. on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, 2021, pp. 987–990; doi:10.1109/ICACCS51430.2021.9441978
17. R. Goyal, N. Chaudhary, and M. Singh, “Machine Learning based Intelligent Career Counselling Chatbot (ICCC),” in *2023 Int. Conf. on Computer Communication and Informatics (ICCCI)*, 2023, pp. 1–8; doi:10.1109/ICCCI56745.2023.10128305
18. V. R. Kumbhar, M. M. Maddel, and Y. Raut, “Smart model for career guidance using hybrid deep learning technique,” in *2023 1st Int. Conf. on Innovations in High-Speed Communication and Signal Processing (IHCSP)*, 2023, pp. 327–331; doi:10.1109/IHCSP56702.2023.10127152
19. C. Yuan, Y. Hong, and J. Wu, “Who Are You Meant to Be? Predicting Psychological Indicators and Occupations based on Personality Traits,” *Journal of Systems Science and Systems Engineering*, vol. 32, no. 5, pp. 571–602, 2023; doi:10.1007/s11518-023-5576-6
20. E. Grunenberg, H. Peters, M. J. Francis, M. D. Back, and S. C. Matz, “Machine learning in recruiting: predicting personality from CVs and short text responses,” *Frontiers in Social Psychology*, vol. 1, p. 1290295, 2024; doi:10.3389/frsps.2023.1290295
21. M. Nirmala et al., “Personality Detection for Recruitment Using Machine Learning,” in *Proc. of the 6th Int. Conf. on Communications and Cyber Physical Engineering, ICCCE 2024*, Singapore: Springer, 2024, pp. 399–406; doi:10.1007/978-981-99-7137-4_38
22. T. Iwasaki, Y. Seki, W. Kashino, A. Keyaki, and N. Kando, “Estimating Citizen Personality Traits Using Social Media Posts,” in *Int. Conf. on Asian Digital Libraries*. Singapore: Springer, 2024, pp. 119–135; doi:10.1007/978-981-96-0868-3_10
23. V. D. Oliseenko, A. O. Khlobystova, A. A. Korepanova, and T. V. Tulupyeva, “Automating the temperament assessment of online social network users,” *Doklady Mathematics*, vol. 108, no. S2, pp. S368–S373, 2023; doi:10.1134/S1064562423701041
24. A. O. Khlobystova, M. V. Abramov, and V. F. Stolyarova, “Research of trends in the relationship between users’ career guidance preferences and their digital footprints in a social network,” *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, vol. 23, no. 3, pp. 564–574, 2023; doi:10.17586/2226-1494-2023-23-3-564-574
25. P. Kiselev, B. Kiselev, V. Matsuta, A. Feshchenko, I. Bogdanovskaya, and A. Kosheleva, “Career guidance based on machine learning: social networks in professional identity construction,” *Procedia Computer Science*, vol. 169, pp. 158–163, 2020; doi:10.1016/j.procs.2020.02.128

Received 20-06-2025, the final version — 15-07-2025.

Anastasiia Ivashchenko, Researcher, Saint Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), ✉ aok@dscs.pro

Arteom Vyatkin, Junior Researcher, Saint Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), aav@dscs.pro

Fedor Bushmelev, Researcher, Saint Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), fvb@dscs.pro

Maxim Abramov, Candidate of Sciences (Tech.), Senior Researcher, Saint Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS), Associate Professor, Saint Petersburg State University, mva@dscs.pro



РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЗИРОВАННОЙ ПОДГОТОВКИ ДОКУМЕНТОВ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ APACHE POI

Мамонов А. А.¹, ассистент, anton.mamonov.golohvastogo@mail.ru

Салпагаров С. И.¹, канд. физ.-мат. наук, доцент, ✉ salpagarov_si@pfur.ru,
orcid.org/0000-0002-5321-9650

Матюшкин Д. В.¹, магистрант, 1032212279@rudn.ru

Миронов Д. А.¹, магистрант, 1032211701@rudn.ru

Кройтор О. К.¹, канд. физ.-мат. наук, старший преподаватель, kroytor_ok@pfur.ru,
orcid.org/0000-0002-5691-7331

¹Российский университет дружбы народов имени Патриса Лумумбы,
ул. Миклухо-Маклая, д. 6, 117198, Москва, Россия

Аннотация

В статье рассматривается разработка программного комплекса для автоматизации документооборота, объединяющего генерацию пакетов документов на основе шаблонов и динамическое создание интерфейсов ввода. Решение реализовано на языке Java с использованием библиотеки Apache POI, оно включает обработку форматов DOC/DOCX, интеграцию с внешними данными (CSV), поддержку многопользовательских сценариев и конвертацию результатов в PDF. Система позволяет устранить ручные операции, минимизировать ошибки форматирования и повысить гибкость взаимодействия с документами. Практическое внедрение на факультете физико-математических и естественных наук РУДН продемонстрировало сокращение временных затрат на 80 % при ежемесячной обработке более 500 документов.

Ключевые слова: автоматизация документооборота, Apache POI, генерация документов, шаблоны документов, CSV-интеграция, конвертация в PDF, динамические интерфейсы, Java, многопользовательские сценарии, обработка DOC/DOCX.

Цитирование: Мамонов А. А., Салпагаров С. И., Матюшкин Д. В., Миронов Д. А., Кройтор О. К. Разработка системы автоматизированной подготовки документов с использованием библиотеки Apache POI // Компьютерные инструменты в образовании. 2025. № 2. С. 48–58. doi:10.32603/2071-2340-2025-2-48-58

1. ВВЕДЕНИЕ

Данная работа посвящена разработке программного комплекса для автоматизации документооборота, объединяющего генерацию пакетов документов на основе шаблонов и динамическое создание интерфейсов ввода. Решение реализовано на языке Java с использованием библиотеки Apache POI, оно включает обработку форматов DOC/DOCX, интеграцию с внешними данными (CSV), поддержку многопользовательских

сценариев и конвертацию результатов в PDF. Основной фокус направлен на устранение ручных операций, минимизацию ошибок и повышение гибкости взаимодействия с документами.

Актуальность автоматизации документооборота остается критически важной задачей для организаций, сталкивающихся с массовой генерацией стандартизированных документов (договоров, отчетов, служебных записок). Ручное заполнение шаблонов не только трудоёмко, но и приводит к ошибкам форматирования, несогласованности данных и задержкам [1]. Существующие системы, такие как ELMA или DMS-платформы, часто требуют ручной адаптации под изменяющиеся условия (например, число авторов или типы данных), сложно интегрируются с внешними источниками и ограничены в поддержке динамического контента. Разрабатываемый комплекс призван решить эти проблемы, предлагая гибкие инструменты для автоматизации, включая динамическую генерацию интерфейсов, интеграцию с CSV-таблицами и многопользовательские шаблоны [2].

Анализ существующих решений включает низкоуровневые методы (Open XML, SDK, OLE Automation, XSLT) и высокоуровневые системы (DMS, ИИ, шаблоны с динамическими полями). Однако они имеют ряд ограничений, представленных в таблице 1 [1, 3]:

- Apache POI требует глубоких технических знаний для настройки и не поддерживает автоматическую адаптацию интерфейсов.
- DMS-системы не обеспечивают динамическое изменение полей ввода в зависимости от числа авторов или структуры данных.
- Решения на основе шаблонов часто сталкиваются с проблемами интеграции с внешними источниками (CSV, SQL) и отсутствием поддержки многопоточности.

Таблица 1. Сравнение систем по ключевым критериям

Критерий	Время генерации	Сохранение форматирования	Многопользовательские шаблоны	Стоимость
Разработанная система	9,96 сек. для Huge-5500	Полное (шрифты, таблицы, стили)	Поддержка через CSV-данные, но без встроенной совместной работы	Бесплатно
Aspose.Words	7,64 сек. для Huge-5500	Полное	Требует внешних интеграций	От \$1199
ELMA	Зависит от интеграции (Aspose быстрее, LibreOffice медленнее)	Возможны искажения при интеграции с LibreOffice	Встроенная поддержка (доступ для нескольких пользователей)	Бесплатная версия с ограничениями и платные лицензии

Кроме того, большинство инструментов не предоставляют механизмов для восстановления разбитых тегов или работы с разными кодировками, что критично для многоязычных сред. Текущие трудности в сфере документооборота включают:

1. Низкую гибкость шаблонов: ручная корректировка под изменяющиеся условия (число авторов, типы данных).
2. Ошибки форматирования: из-за ручного ввода и дублирования контента.
3. Сложности интеграции: несовместимость форматов данных (DOCX, CSV, SQL) и кодировок.

4. Ограниченную масштабируемость: невозможность работы с большими объемами документов в многопользовательском режиме.
5. Отсутствие стандартизации: разнородные интерфейсы для ввода данных, требующие дополнительного обучения пользователей.

Методология исследования основана на объектно-ориентированном подходе с использованием паттерна MVC. Для обработки документов применена библиотека Apache POI [4], а для хранения данных — SQLite [5]. Система автоматически генерирует интерфейс на основе шаблонов, поддерживая работу с несколькими авторами. Реализованы парсинг CSV-таблиц с обработкой кириллицы cp-1251 [6] и конвертация в PDF через documents4j [7]. Тестирование включало проверку производительности и корректности обработки данных.

2. ОПИСАНИЕ РАЗРАБОТКИ СИСТЕМЫ

Благодаря реализации программных решений для работы с документами Microsoft Office, современные системы документооборота обрели способность автоматизировать процесс создания документов на основе шаблонов. Несмотря на то, что существуют различные инструменты для работы с документами Office, такие как Microsoft Office Interop, OpenOffice API и другие, особое место среди подобных решений занимает библиотека Apache POI. Она поддерживает как файлы .doc, представляющие собой бинарный формат Microsoft Word [8], так и файлы .docx, основанные на стандарте Office Open XML [9] и представляющие собой ZIP-архив с набором XML-файлов. Помимо базовой функциональности работы с документами, Apache POI предоставляет богатый набор API для создания, чтения, редактирования и анализа различных компонентов документов, включая текст, таблицы, формулы и графические элементы. Благодаря этой универсальности и обратной совместимости, Apache POI широко используется в корпоративных системах для автоматизации документооборота, генерации отчетов и массовой обработки документов различных форматов.

В течение последних лет библиотека Apache POI активно развивалась, создавая новые возможности для оптимизации работы с документами. Это развитие включало улучшение производительности, добавление поддержки новых форматов документов и расширение функциональности для работы с макросами и формулами, что сделало библиотеку еще более мощным инструментом для разработчиков.

Результатом этого развития стала версия 5.2.5, предоставляющая широкий набор инструментов для работы с различными форматами документов. В этой версии были существенно улучшены механизмы обработки больших файлов, добавлена расширенная поддержка стилей и форматирования, а также внедрены новые API для более удобной работы с электронными таблицами и презентациями. Однако тестирование и внедрение данной версии требует создания специализированных решений, учитывающих особенности работы с различными форматами документов и спецификой их использования в реальных условиях.

На рисунке 1 можно видеть схематическое представление процесса обработки документа в разработанной системе, включающее этапы загрузки, анализа структуры, обработки содержимого документа и конвертации в PDF.

При разработке системы автоматизированной подготовки документов одним из первых этапов являлась интеграция библиотеки Apache POI через систему управления зависимостями Maven. Интеграция была реализована путем добавления соответствующих зависимостей: для работы с форматами .docx и .doc используются артефакты poi-ooxml

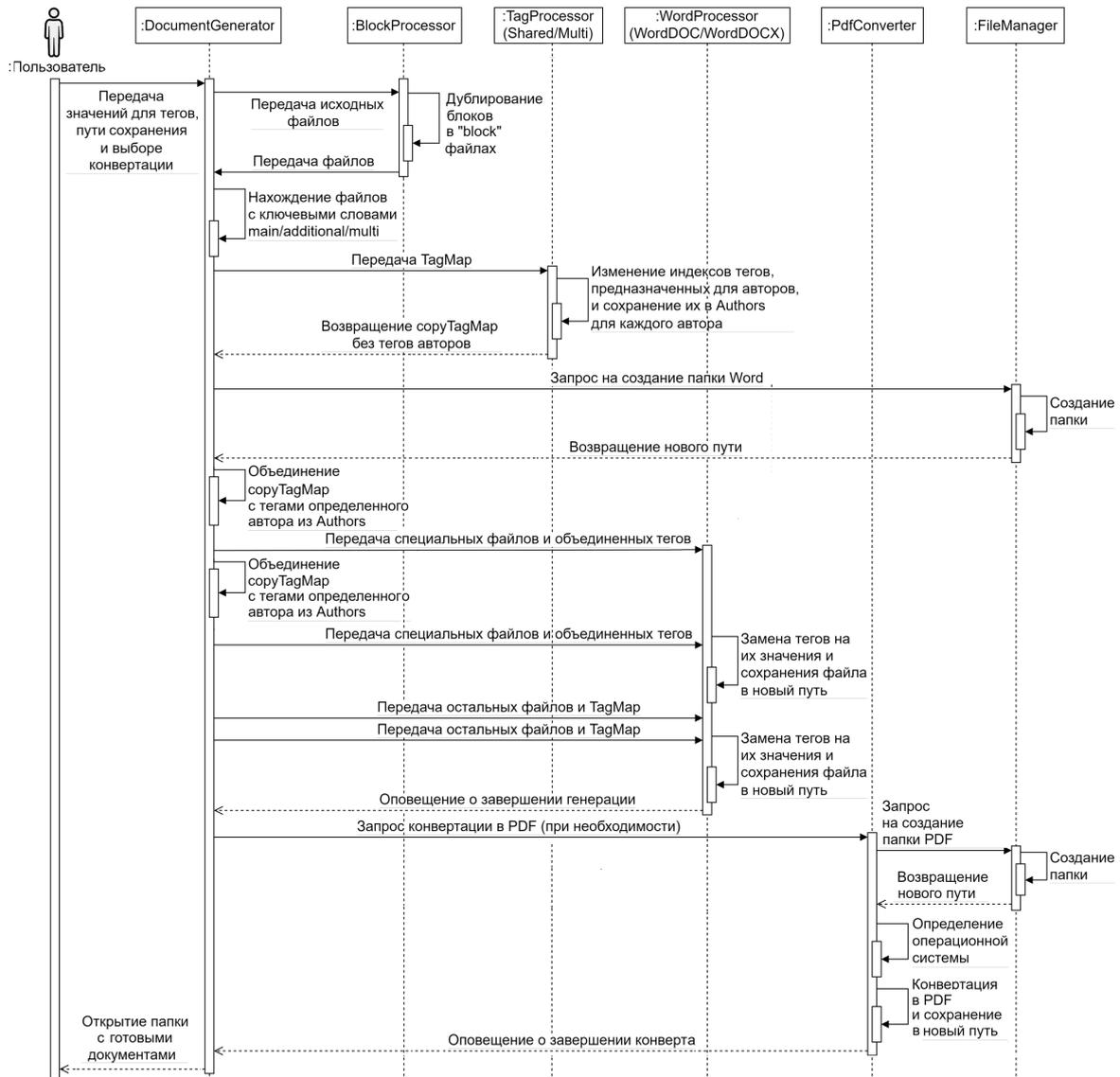


Рис. 1. Диаграмма последовательности

и роi-scratchpad соответственно. Это позволяет интегрировать библиотеку в проект и начать использование её функциональности для обработки документов.

Использование Maven в качестве системы управления зависимостями обусловлено такими его преимуществами как: автоматизированное управление зависимостями, стандартизированная структура проекта, простота развертывания и обновления компонентов. Кроме того, Maven обеспечивает централизованное хранение артефактов в репозиториях, поддерживает управление версиями, автоматическую сборку проекта и интеграцию с системами непрерывной интеграции, что значительно упрощает процесс разработки и поддержки проекта в долгосрочной перспективе.

Для реализации системы была разработана диаграмма классов (рис. 2), которая показывает внутреннюю организацию приложения, его основные сущности и взаимосвязи между ними. Данная диаграмма позволяет не только понять динамику работы системы, но и увидеть, как её компоненты организованы на структурном уровне.

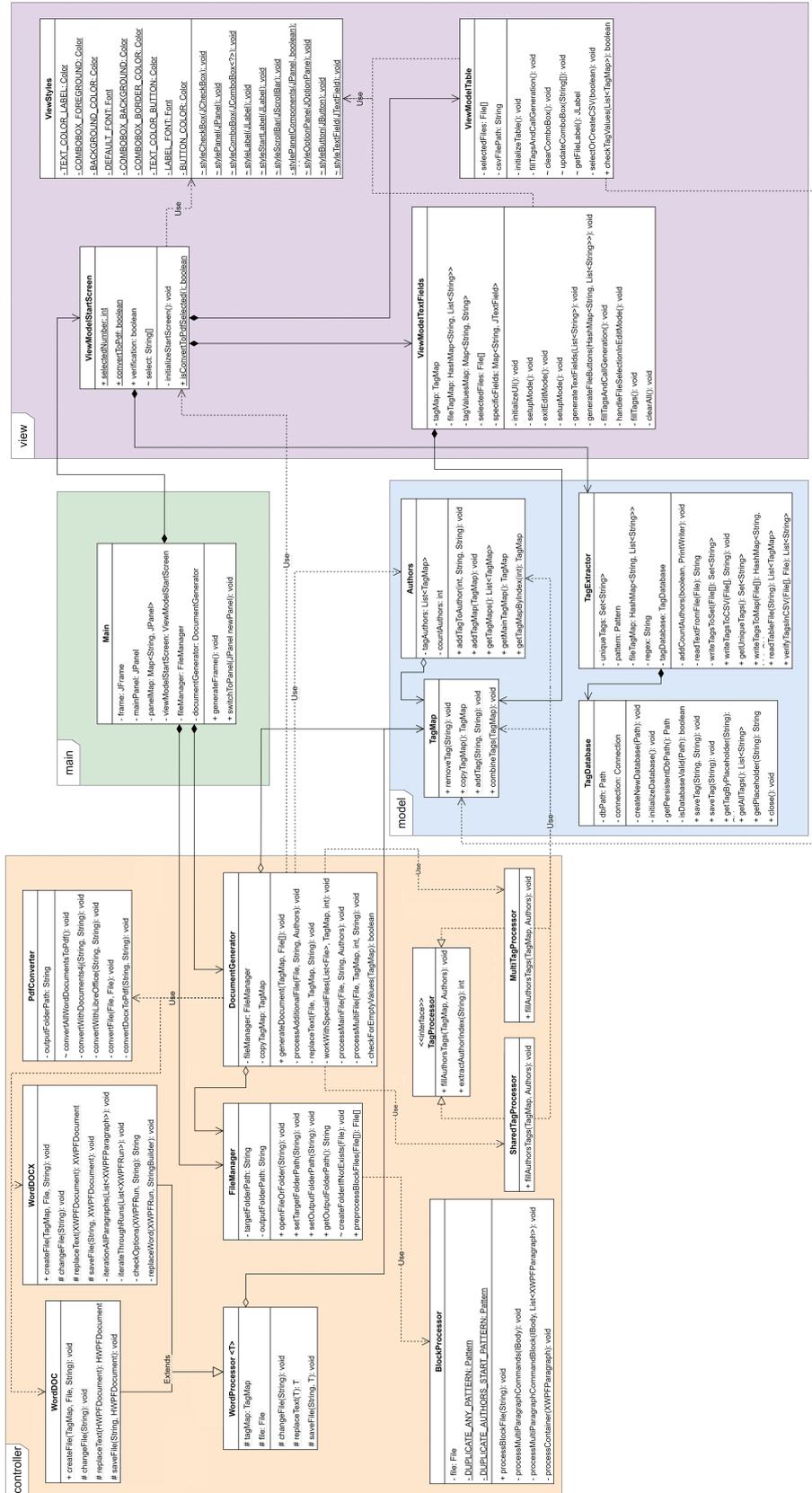


Рис. 2. Диаграмма классов

В основе такой организации лежит паттерн Model–View–Controller (MVC), обеспечивающий чёткое разделение логики, интерфейса и управления данными:

1. **Model.** Отвечает за хранение и структуру данных — в данном случае классы, связанные с хранением информации о тегах и авторах.
2. **View.** Представляет собой визуальную часть приложения (окна, формы).
3. **Controller.** Содержит основную бизнес-логику и координирует работу системы.
4. **Main.** Содержит точку входа в приложение и инициализирует контроллеры, модель и представление.

Благодаря такому распределению классов по слоям и чётким зонам ответственности упрощается поддержка и расширение приложения: новые типы тегов или форматов документов можно добавлять в соответствующие контроллеры и модели без затрагивания других компонентов.

Для обеспечения корректной обработки документов была реализована универсальная процедура обработки текстовых меток (тегов), параметризованная типом документа и способом его обработки. Границы обработки задаются структурой документа и спецификой используемого формата, что позволяет гибко конфигурировать процесс подготовки документов в соответствии с конкретными требованиями.

- **Стартовый экран** (рис. 3) позволяет задать базовые параметры документа: количество авторов, путь сохранения результатов, необходимость конвертации в PDF. Выпадающий список методов генерации (*Использовать поля ввода*, *Использовать таблицу*) определяет дальнейший workflow. Интерфейс реализует прогрессивное раскрытие функций — например, опция *Редактировать подсказки* активируется только после выбора шаблона.

RUDN University

Выберите количество авторов

3 ▼

Выбрать папку для сохранения

Путь: C:\Users\Admin\Documents\New Folder

Конвертировать в .pdf?

Как сгенерировать документ?

Использовать поля ввода

Использовать таблицу

Редактировать подсказки

Сбросить состояние системы

Рис. 3. Стартовое окно

- **Генерация через CSV-таблицы** (рис. 4) предоставляет инструменты для автоматизированного заполнения данных. Пользователь может создать новую таблицу, загрузить существующую или выбрать файлы шаблонов (.doc/docx). Поддержка динамического связывания полей (например, *Обложка для диска.docx*) с колонками CSV обеспечивает пакетную обработку документов. Алгоритм валидации предотвращает конфликт типов данных и форматирования.

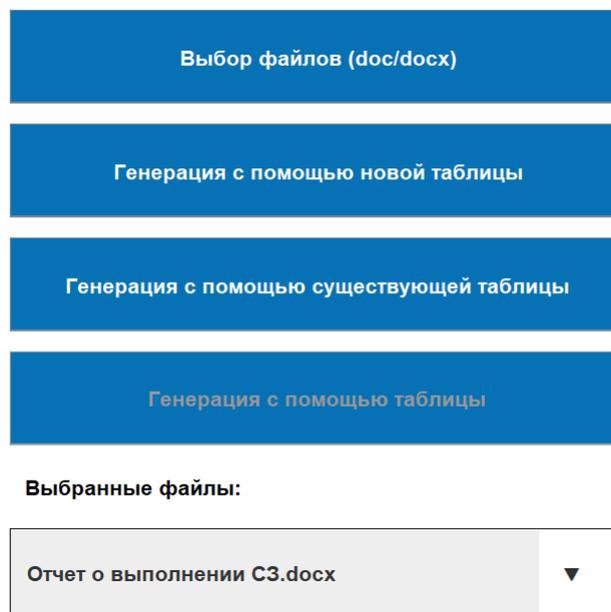


Рис. 4. Окно генерации с помощью CSV таблицы

- **Ручной ввод через текстовые поля** (рис. 5) реализует контекстно-зависимую форму с примерами заполнения (*Индекс, Страна, Город...*). Система автоматически генерирует поля на основе тегов шаблона ($\$(key_ria_author1_address)$), поддерживая валидацию в реальном времени. Вкладка *Показать все теги* отображает полный список заменяемых элементов, упрощая навигацию для сложных документов.

Следует отметить, что исходные файлы данной системы размещены в открытом доступе и могут быть загружены из репозитория на платформе GitHub по адресу: <https://github.com/stifell/template-process/tree/master>.

3. ТЕСТИРОВАНИЕ И РЕЗУЛЬТАТЫ

Был проведен анализ времени генерации документов с использованием набора тестов, реализованных с помощью JUnit [10]. Тестирование проводилось на персональном компьютере с процессором AMD Ryzen 5 7000 Series, 16 ГБ оперативной памяти и операционной системой Windows 11. Для оценки производительности использовалась среда Java SE 21 с библиотекой Apache POI 5.2.5. Цель тестирования — изучить влияние объема текста и количества тегов на производительность системы, а также оценить время дополнительной конвертации документов в PDF. Результаты представлены в таблице 2.

При сравнении файлов с одинаковым объемом текста увеличение количества тегов приводит к существенному росту времени генерации. Например, для файлов с 50000 сло-

Рис. 5. Окно для заполнения документов с помощью текстовых полей

вами среднее время генерации увеличивается с 0,12864 секунд (тип Huge-1500) до 9,95829 секунд (тип Huge-5500). Аналогичная тенденция наблюдается и для файлов с 10000 словами — время возрастает с 0,09935 секунд (Medium-1500) до 1,61678 секунд (Medium-5500).

При фиксированном количестве тегов увеличение объема текста оказывает менее заметное влияние на время генерации. Так, для файлов с 1500 тегами увеличение количества слов с 10000 (Medium-1500) до 50000 (Huge-1500) приводит к росту времени генерации лишь с 0,09935 до 0,12864 секунд. Дополнительное время, затрачиваемое на конвертацию документов в PDF, включает в себя время генерации документа. Разница между временем генерации и временем конвертации показывает, что влияние объема текста на этот процесс более заметно, что обусловлено дополнительными операциями преобразования формата.

Таблица 2. Сводная таблица тестовых данных

Тип файла	Слов в файле	Количество файлов	Тегов на файл	Среднее время на файл (с)	Среднее время с конвертацией (с)
Small-1500	2000	100	1500	0,07305	2,83005
Medium-1500	10000	50	1500	0,09935	3,16758
Huge-1500	50000	10	1500	0,12864	7,06854
Medium-2600	10000	50	2600	0,21539	3,27142
Huge-2600	50000	10	2600	1,37381	8,51631
Medium-5500	10000	50	5500	1,61678	5,07708
Huge-5500	50000	10	5500	9,95829	17,25088

На основании проведённого тестирования можно сделать вывод, что основным фактором, определяющим время генерации документов, является именно количество тегов, внедряемых в шаблон. При этом объём текста оказывает менее значительное, но всё же ощутимое влияние.

4. ЗАКЛЮЧЕНИЕ

В результате выполнения данного проекта была разработана система автоматизированной подготовки документов на базе библиотеки Apache POI. Эта система успешно решает ключевые проблемы современных инструментов документооборота, предлагая универсальное и надёжное решение. Благодаря поддержке форматов DOC и DOCX с сохранением структурной целостности, включая таблицы, формулы и графические элементы, система демонстрирует высокую адаптивность к разнородным требованиям. Интеграция с внешними источниками, такими как CSV-таблицы, с обработкой кириллических кодировок (cp-1251), обеспечивает взаимодействие с существующими корпоративными системами, устраняя необходимость ручного переноса информации.

Практическое внедрение на факультете физико-математических и естественных наук РУДН продемонстрировало сокращение временных затрат на 80 % при ежемесячной обработке более 500 документов, что подчеркивает ее готовность к использованию в реальных условиях. Особую ценность представляет многопользовательский режим, позволяющий командам совместно работать над сложными проектами без риска конфликтов версий.

Перспективы развития системы связаны с расширением функциональности через интеграцию ИИ-моделей для семантического анализа контента и автоматической генерации шаблонов, что сократит время настройки под специфические задачи. Внедрение блокчейн-технологий для верификации цепочки изменений усилит безопасность в сценариях с множеством участников, а поддержка облачных провайдеров (Google Drive, Yandex Disk) через REST API упростит распределенную работу с документами. Оптимизация кэширования для обработки экстремально больших файлов (1 млн + слов) и разработка кросс-платформенного веб-клиента сделают систему ещё более универсальной. Эти улучшения не только укрепят позиции решения на рынке, но и заложат основу для интеллектуальных систем следующего поколения, способных трансформировать подходы к управлению документами в эпоху цифровой трансформации.

Список литературы

1. Павлов А. К. Обзор методов автоматизации разработки документов в организации // Международный журнал гуманитарных и естественных наук. 2024. № 6–3 (93). С. 195–199. doi:10.24412/2500-1000-2024-6-3-195-199
2. Леонова М. В. Интерактивные интерфейсы для автоматизации документооборота. Современные технологии документооборота в бизнесе, производстве и управлении // Сборник материалов XXIV Всероссийской научно-практической конференции. Москва, 2024. М.: Абрис, 2024. С. 87–95.
3. Кравченко В. В. Сравнительный анализ методов генерации документов формата docx по шаблону // Математическое и программное обеспечение информационных, технических и экономических систем. Томск: Издательский Дом Томского государственного университета, 2013. С. 15–17.
4. Apache Software Foundation. Apache poi javadocs, 01 2025 [Электронный ресурс]. URL: <https://poi.apache.org/> (дата обращения: 14.01.2025).

5. D. Richard Hipp. Sqlite documentation, 2023 [Электронный ресурс]. URL: <https://www.sqlite.org/docs.html> (дата обращения: 15.04.2025).
6. Unicode Consortium. Code page 1251 encoding, 2023 [Электронный ресурс]. URL: <https://www.unicode.org/> (дата обращения: 15.04.2025).
7. documents4j Team. documents4j — document conversion api, 2023 [Электронный ресурс]. URL: <https://documents4j.com/> (дата обращения: 15.04.2025).
8. Microsoft Learn. [ms-doc]: Word (.doc) binary file format [Электронный ресурс]. URL: <https://learn.microsoft.com/> (дата обращения: 20.03.2025).
9. ECMA International. ECMA-376 Office Open XML File Formats, fifth edition, 2021 [Электронный ресурс]. URL: <https://www.ecma-international.org/> (дата обращения: 15.03.2025).
10. JUnit Team. Junit 5 user guide [Электронный ресурс]. URL: <https://junit.org/junit5/> (дата обращения: 17.04.2025).

Поступила в редакцию 19.05.2025, окончательный вариант — 15.07.2025.

Мамонов Антон Алексеевич, ассистент, Российский университет дружбы народов им. Патриса Лумумбы, anton.mamonov.golohvastogo@mail.ru

Салпагаров Солтан Исмаилович, канд. физ.-мат. наук, доцент, Российский университет дружбы народов им. Патриса Лумумбы, ✉ salpagarov_si@pfur.ru

Матюшкин Денис Владимирович, магистрант, Российский университет дружбы народов им. Патриса Лумумбы, 1032212279@rudn.ru

Миронов Дмитрий Андреевич, магистрант Российский университет дружбы народов им. Патриса Лумумбы, 1032211701@rudn.ru

Кройтор Олег Константинович, канд. физ.-мат. наук, старший преподаватель, Российский университет дружбы народов им. Патриса Лумумбы, kroytor_ok@pfur.ru

Computer tools in education, 2025

№ 2: 48–58

<http://cte.eltech.ru>

[doi:10.32603/2071-2340-2025-2-48-58](https://doi.org/10.32603/2071-2340-2025-2-48-58)

Development of an Automated Document Preparation System Using the Apache POI Library

Mamonov A. A.¹, Assistant, anton.mamonov.golohvastogo@mail.ru

Salpagarov S. I.¹, Cand. Sc., Associate Professor, ✉ salpagarov_si@pfur.ru,
orcid.org/0000-0002-5321-9650

Matyushkin D. V.¹, Master's Degree student, 1032212279@rudn.ru

Mironov D. A.¹, Master's Degree student, 1032211701@rudn.ru

Kroytor O. K.¹, Cand. Sc., Senior Lecturer, kroytor_ok@pfur.ru, orcid.org/0000-0002-5691-7331

¹RUDN University, 6 Miklukho-Maklaya str., 117198, Moscow, Russia

Abstract

The article discusses the development of a software system for document workflow automation, combining the generation of document packages based on templates and dynamic creation of input interfaces. The solution is implemented in Java using the Apache

POI library, provides processing of DOC / DOCX formats, integration with external data (CSV), support for multi-user scenarios, and conversion of results to PDF. The system eliminates manual operations, minimizes formatting errors, and increases flexibility in document interaction. Practical implementation at the Faculty of Physics, Mathematics and Natural Sciences of RUDN University demonstrated an 80 % reduction in time costs when processing more than 500 documents monthly.

Keywords: *document workflow automation, Apache POI, document generation, document templates, CSV integration, PDF conversion, dynamic interfaces, Java, multi-user scenarios, DOC / DOCX processing.*

Citation: A. A. Mamonov, S. I. Salpagarov, D. V. Matyushkin, D. A. Mironov, and O. K. Kroytor, "Development of an Automated Document Preparation System Using the Apache POI Library," *Computer tools in education*, no. 2, pp. 48–58, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-48-58

References

1. A. K. Pavlov, "Overview of automation methods for document development in an organization," *International Journal of Humanities and Natural Sciences*, no. 6-3 (93), pp. 195–199, 2024 (in Russian); doi:10.24412/2500-1000-2024-6-3-195-199
2. M. V. Leonova, "Interactive interfaces for document workflow automation," in *Proc. of Modern Document Management Technologies in Business, Production and Management: XXIV All-Russian Sci. Pract. Conf.*, Moscow: Abris Publishing, , pp. 87–95, 2024 (in Russian).
3. V. V. Kravchenko, "Comparative analysis of methods for generating docx format documents from a template," in *Mathematical and Software Support for Information, Technical and Economic Systems*, Tomsk, Russia: Tomsk State Univ. Press, , pp. 15–17, 2013 (in Russian).
4. "Apache Software Foundation", in *Apache POI Javadocs*, 2025. [Online]. Available: <https://poi.apache.org/>
5. D. R. Hipp, *SQLite Documentation*, 2023. [Online]. Available: <https://www.sqlite.org/docs.html>
6. "Unicode Consortium," in *Code Page 1251 Encoding*, 2023. [Online]. Available: <https://www.unicode.org/>
7. "documents4j Team," in *documents4j — Document Conversion API*, 2023. [Online]. Available: <https://documents4j.com/>
8. "Microsoft Learn," in *[MS-DOC]: Word (.doc) Binary File Format*, 2025. [Online]. Available: <https://learn.microsoft.com/>
9. "ECMA International," in *ECMA-376 Office Open XML File Formats*, 5th ed., 2021. [Online]. Available: <https://www.ecma-international.org/>
10. "JUnit Team," in *JUnit 5 User Guide*, 2025. [Online]. Available: <https://junit.org/junit5/>

Received 19-05-2025, the final version — 15-07-2025.

Anton Mamonov, Assistant, Friendship University of Russia (RUDN University),
anton.mamonov.golohvastogo@mail.ru

Soltan Salpagarov, Cand. of Sciences (Phys.-Math.), Associate Professor, Friendship University of Russia (RUDN University), ✉ salpagarov_si@pfur.ru

Denis Matyushkin, Master's Degree student, Friendship University of Russia (RUDN University),
1032212279@rudn.ru

Dmitry Mironov, Master's Degree student, Friendship University of Russia (RUDN University) ,
1032211701@rudn.ru

Oleg Kroytor, Cand. of Sciences (Phys.-Math.), Senior Lecturer, Friendship University of Russia (RUDN University), kroytor_ok@pfur.ru



СОЗДАНИЕ И ПРИМЕНЕНИЕ ЗАДАНИЙ ФОРМАТА «ПРОПУСКИ» В ОНЛАЙН-КУРСАХ НА ПЛАТФОРМЕ STEPİK

Багоутдинова А. Г.¹, канд. техн. наук, доцент, bagoutdinova@rambler.ru

¹Казанский федеральный университет, ул. Кремлевская, 35, 420008, Казань, Россия

Аннотация

В статье рассмотрен опыт создания и применения заданий формата «Пропуски» в онлайн-курсах «Линейная алгебра и аналитическая геометрия» и «Теория вероятностей и математическая статистика». Данные курсы были специально разработаны как дополнение к аудиторным занятиям для студентов Института управления, экономики и финансов Казанского федерального университета и размещены на образовательной платформе Stepik. Значительная часть практических задач в этих курсах представлена в формате «Пропуски», который весьма интересен. Дело в том, что одно тестовое задание может включать несколько вопросов, чередующихся с текстом. Такой формат позволяет не только оценить конечный ответ студента, но и проследить весь ход его рассуждений при решении задачи.

В работе представлены конкретные примеры из курсов, иллюстрирующие возможности заданий с пропусками, а также даются рекомендации по их эффективному применению. Эмпирические данные подтверждают положительный эффект от внедрения курсов, включая повышение среднего балла на 9,5 % и сокращение доли неудовлетворительных оценок по темам, которые ранее вызывали затруднения.

Статья будет полезна педагогам, заинтересованным в разработке эффективных заданий для повышения качества образования и мотивации обучающихся в условиях онлайн-формата или гибридного обучения.

Ключевые слова: задания формата «Пропуски», образовательная платформа Stepik, онлайн-курсы по математике.

Цитирование: Багоутдинова А. Г. Создание и применение заданий формата «пропуски» в онлайн-курсах на платформе Stepik // Компьютерные инструменты в образовании. 2025. № 2. С. 59–72. doi:10.32603/2071-2340-2025-2-59-72

1. ВВЕДЕНИЕ

В современном образовательном пространстве онлайн-курсы играют важную роль не только как инструменты самостоятельного освоения дисциплин, но и как эффективное дополнение к традиционному обучению. Они могут выполнять функцию цифровых учебников, предоставляя теоретические материалы и позволяя студентам отрабатывать практические навыки в интерактивном режиме с автоматизированной проверкой решений для самоконтроля. Сочетание традиционных форм обучения (лекций, семинаров) с самостоятельной работой в онлайн-режиме способствует глубокому усвоению материала.

ла, а преподаватели получают возможность мониторить академическую активность студентов и анализировать данные для корректировки учебного процесса.

В интернете доступно множество платформ для создания и поддержки онлайн-курсов, таких как Moodle, Coursera, Udacity Google Classroom, и др. Одной из наиболее популярных и удобных платформ является российская образовательная платформа Stepik [1, 2]. Она обладает интуитивным интерфейсом, широким набором типов заданий и системой отслеживания прогресса. Функционал Stepik позволяет преподавателям создать контент, который не только проверяет итоговый ответ, но и контролирует каждый этап решения, что критически важно для формирования осознанного понимания материала. Среди разнообразных типов заданий на платформе особое место занимают задания с пропусками.

Задания типа «Пропуски» получили широкое распространение в онлайн-образовании и зарекомендовали себя как эффективный формат организации учебного процесса. Изначально такие задания разрабатывались для изучения иностранных языков, где они активно используются для оценки уровня владения грамматикой, словарным запасом и пониманием контекста [3, 4]. Однако область применения заданий «Пропуски» не ограничивается лингвистикой. Они оказываются эффективными и в других дисциплинах: математике, информатике, физике, химии, медицине и т. п. [5–9]. Несмотря на широкое распространение формата и платформы, в научно-методической литературе наблюдается дефицит работ, обобщающих практический опыт создания и применения заданий формата «Пропуски» именно на платформе Stepik для преподавания математических дисциплин. Преподаватели-практики редко публикуют детальные описания реализации такого подхода и анализ его эффективности. Настоящая статья призвана восполнить этот пробел.

Цель данной статьи — обобщить опыт создания и применения заданий формата «Пропуски» в онлайн-курсах «Линейная алгебра и аналитическая геометрия» [10] и «Теория вероятностей и математическая статистика» [11], разработанных для студентов направления «Экономика» Института управления, экономики и финансов Казанского федерального университета, а также оценить их эффективность на основе анализа академических результатов студентов.

В качестве основных методов исследования были использованы:

- Теоретический анализ научно-методической литературы по проблеме исследования.
- Обобщение педагогического опыта разработки и внедрения онлайн-курсов.
- Сравнительный анализ академической успеваемости студентов до и после внедрения курсов.
- Педагогическое наблюдение за вовлеченностью студентов в учебный процесс.

Курсы созданы в полном соответствии с рабочей программой дисциплины «Математика» и размещены на образовательной платформе Stepik. Студенты использовали материалы курса для выполнения домашних заданий, подготовки к контрольным работам и экзаменам. Для обучающихся, которые не могли регулярно посещать очные занятия, эти ресурсы стали основным инструментом самостоятельного освоения дисциплины.

Такой подход позволил эффективно организовать учебный процесс, обеспечив оптимальное распределение времени: на аудиторных занятиях под руководством преподавателя детально разбираются типовые задачи, а во внеурочное время студенты закрепляют материал через самостоятельную онлайн-отработку практических заданий. Практические задания выстроены по принципу постепенного усложнения: от элементарных за-

дач с подробными инструкциями и подсказками — к сложным, требующим полной самостоятельности в решении.

В статье представлены конкретные примеры заданий формата «Пропуски», применяемых в онлайн-курсах по математическим дисциплинам. Отмечено, что подобные задания способствуют лучшему усвоению материала и помогают студентам развивать внимание к ключевым аспектам изучаемых тем. Кроме того, в работе предлагаются практические рекомендации, которые помогут оптимизировать процесс разработки и внедрения таких заданий в онлайн-курсы, достигая максимального эффекта от их использования.

Материалы статьи окажутся полезными для преподавателей, работающих в сфере онлайн-обучения и заинтересованных в разработке и внедрении инновационных методов преподавания, нацеленных на повышение мотивации студентов и улучшение качества образовательного процесса.

2. ПРИМЕРЫ ЗАДАНИЙ ИЗ КУРСА «ЛИНЕЙНАЯ АЛГЕБРА И АНАЛИТИЧЕСКАЯ ГЕОМЕТРИЯ»

На платформе Stepik доступно 18 типов заданий с автоматической или ручной проверкой [12]. В данной статье основное внимание уделяется формату «Пропуски», где обучающийся заполняет пропуски в тексте, вводя ответ с клавиатуры или выбирая его из выпадающего списка.

Ключевые особенности формата:

- **автоматическая обратная связь:** неверные ответы помечаются красным крестом, верные — зеленой галочкой;
- **гибкая система оценивания:** возможно получение частичного балла. Баллы распределяются пропорционально числу правильно заполненных пропусков.

Раздел «Линейная алгебра и аналитическая геометрия» изучается студентами направления «Экономика» Института управления, экономики и финансов Казанского федерального университета во втором семестре в рамках курса «Математика». Для успешного освоения дисциплины и отработки практических навыков разработан онлайн-курс на платформе Stepik, значительная часть заданий которого выполнена в формате «Пропуски».

Далее приведены примеры заданий данного формата, наглядно показывающие его достоинства.

Задача 1. Вычислите произведение матриц

$$\begin{pmatrix} -1 & -3 & 0 \\ 2 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} -1 & 0 \\ 3 & 4 \\ 1 & -2 \end{pmatrix}.$$

Реализация на Stepik. Задача разбивается на отдельные шаги — вычисление каждого элемента итоговой матрицы. Для каждого шага создаётся отдельный пропуск, в который студент вводит промежуточный результат (рис. 1). Цель задания: отработать алгоритм умножения матриц пошагово.

Шаг 1: Вычисление элементов первой строки матрицы:

$$c_{11} = (-1) \cdot (-1) + (-3) \cdot 3 + 0 \cdot 1 = [\text{пропуск}],$$

$$c_{12} = (-1) \cdot 0 + (-3) \cdot 4 + 0 \cdot (-2) = [\text{пропуск}].$$

Шаг 2: Вычисление элементов второй строки матрицы:

$$c_{21} = 2 \cdot (-1) + 0 \cdot 3 + 1 \cdot 1 = [\text{пропуск}],$$

$$c_{22} = 2 \cdot 0 + 0 \cdot 4 + 1 \cdot (-2) = [\text{пропуск}].$$

**Линейная алгебра и
аналитическая геометрия**
Прогресс по курсу: 292/334

2 Матрицы

2.1 Виды матриц

2.2 Действия с матрицами

2.3 Умножение матриц

2.4 Элементарные преобр...

2.5 Интересные задачи

3 Определители

3.1 Определители первог...

3.2 Определители n-го по...

3.3 Свойства определите...

3.4 Вычисление определе...

Вычислите **произведение** матриц

$$C = \begin{pmatrix} -1 & -3 & 0 \\ 2 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} -1 & 0 \\ 3 & 4 \\ 1 & -2 \end{pmatrix}.$$

Заполните пропуски

Первая строка матрицы C:

$c_{11} = (-1) \cdot (-1) + (-3) \cdot 3 + 0 \cdot 1 =$

$c_{12} = (-1) \cdot 0 + (-3) \cdot 4 + 0 \cdot (-2) =$

Вторая строка матрицы C:

$c_{21} = 2 \cdot (-1) + 0 \cdot 3 + 1 \cdot 1 =$

$c_{22} = 2 \cdot 0 + 0 \cdot 4 + 1 \cdot (-2) =$

Рис. 1. Задание на умножение матриц формата «Пропуски» с пошаговой проверкой элементов

Каждый пропуск автоматически проверяется системой. Студент видит, какие элементы матрицы вычислены неверно, и может исправить ошибки. Завершив предыдущее задание, студент приступает к следующему (рис. 2), где вводится вся матрица C целиком, без подсказок и промежуточной проверки, что способствует закреплению навыка.

**Линейная алгебра и
аналитическая геометрия**
Прогресс по курсу: 292/334

2 Матрицы

2.1 Виды матриц

2.2 Действия с матрицами

2.3 Умножение матриц

2.4 Элементарные преобр...

2.5 Интересные задачи

3 Определители

3.1 Определители первог...

3.2 Определители n-го по...

Вычислите **произведение** матриц

$$C = \begin{pmatrix} 1 & -3 & 2 \\ 2 & 0 & 1 \\ 0 & 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} -1 & 0 \\ 2 & 4 \\ 1 & -2 \end{pmatrix}.$$

Заполните пропуски

$c_{11} =$ $c_{12} =$

$c_{21} =$ $c_{22} =$

$c_{31} =$ $c_{32} =$

Рис. 2. Задание на умножение матриц без промежуточных подсказок

Приведем ещё один пример, иллюстрирующий поэтапное освоение метода Крамера. На начальном этапе студентам предлагается решить простую систему линейных уравнений размером 2×2 . Каждый шаг решения сопровождается подсказками: сначала рассчитываются определители, затем находятся значения искоемых переменных.

Задача 2. Решите систему линейных уравнений методом Крамера

$$\begin{cases} 2x_1 - x_2 = -7, \\ x_1 + 3x_2 = 7. \end{cases}$$

Реализация на Stepik. Решение разбито на шаги (рис. 3).

Шаг 1: Вычисление определителей:

$$\Delta = \begin{vmatrix} 2 & -1 \\ 1 & 3 \end{vmatrix} = [\text{пропуск}],$$

$$\Delta_1 = \begin{vmatrix} -7 & -1 \\ 7 & 3 \end{vmatrix} = [\text{пропуск}],$$

$$\Delta_2 = \begin{vmatrix} 2 & -7 \\ 1 & 7 \end{vmatrix} = [\text{пропуск}].$$

Шаг 2: Вычисление неизвестных по формулам Крамера:

$$x_1 = \frac{\Delta_1}{\Delta},$$

$$x_2 = \frac{\Delta_2}{\Delta}.$$

The screenshot shows a course interface for 'Линейная алгебра и аналитическая геометрия'. The progress is 292/334. The current section is '5 Матричные уравнения', specifically '5.4 Решение систем линейных уравнений'. The task is to solve a system of linear equations using Cramer's method. The system is:
$$\begin{cases} 2x_1 - x_2 = -7, \\ x_1 + 3x_2 = 7. \end{cases}$$
The interface prompts the user to 'Заполните пропуски' (fill in the blanks). It shows the calculation of determinants:
$$\Delta = \begin{vmatrix} 2 & -1 \\ 1 & 3 \end{vmatrix} = \text{[input field]}$$

$$\Delta_1 = \begin{vmatrix} -7 & -1 \\ 7 & 3 \end{vmatrix} = \text{[input field]}$$

$$\Delta_2 = \begin{vmatrix} 2 & -7 \\ 1 & 7 \end{vmatrix} = \text{[input field]}$$
Then, it prompts for the solution:
$$x_1 = \frac{\Delta_1}{\Delta} = \text{[input field]}$$

$$x_2 = \frac{\Delta_2}{\Delta} = \text{[input field]}$$

Рис. 3. Решение системы уравнений методом Крамера с подсказками

Для закрепления навыков применения метода Крамера и углубления понимания его алгоритма студентам предлагается решить систему линейных уравнений увеличенной

размерности (например, 3×3). Уровень помощи сокращается: студенты решают задачу самостоятельно, но получают подсказки относительно основных этапов алгоритма (рис. 4).

На заключительном этапе студентам предлагается задача на решение системы линейных уравнений без промежуточных подсказок. Требуется вводить только итоговые значения переменных. Оценивается исключительно конечный результат, завершая цикл обучения, поэтапно формируя у студентов способность к самостоятельному решению задач и полному овладению методом.

Задания формата «Пропуски» на платформе Stepik предоставляют гибкость выбора способа ввода ответа благодаря двум основным режимам: «Ввод текста» и «Выбор». Режим «Выбор» автоматизирует проверку заданий с заранее определённым набором корректных ответов. Режим «Ввод текста» используется для открытых вопросов, где ответ необходимо сформулировать самостоятельно.

Комбинируя оба режима, можно создавать разнообразные задания разных уровней сложности, делая обучение более увлекательным и продуктивным.

Совместное применение обоих режимов в одном задании показано на рис. 5.

Линейная алгебра и аналитическая геометрия
Прогресс по курсу: 292/334

5 Матричные уравнения

5.1 Обратная матрица

5.2 Решение матричных у...

5.3 Применение обратной...

5.4 Решение систем линейных уравнений

5.5 Применение матриц к ...

5.6 Интересные задачи

6 Системы линейных алге...

6.1 Общие сведения

6.2 Исследование систем...

6.3 Метод Гаусса

6.4 Базисные решения

Решите систему линейных уравнений методом Крамера

$$\begin{cases} x_1 - 3x_2 + 2x_3 = -9, \\ x_1 + x_2 + 3x_3 = 4, \\ 2x_1 + x_2 - x_3 = -2. \end{cases}$$

Заполните пропуски

Вычислим определители:

$\Delta =$ $\Delta_1 =$

$\Delta_2 =$ $\Delta_3 =$

Решение системы уравнений:

$x_1 =$

$x_2 =$

$x_3 =$

Рис. 4. Решение системы уравнений 3×3 методом Крамера с сокращёнными подсказками

3. ПРИМЕРЫ ЗАДАНИЙ ИЗ КУРСА «ТЕОРИЯ ВЕРОЯТНОСТЕЙ И МАТЕМАТИЧЕСКАЯ СТАТИСТИКА»

«Теория вероятностей и математическая статистика» изучается студентами направления «Экономика» Института управления, экономики и финансов Казанского федерального университета в третьем семестре в рамках курса «Математика». Для повышения эффективности обучения и развития практических навыков на платформе Stepik был разработан одноимённый онлайн-курс, в котором задания формата «Пропуски» активно

Линейная алгебра и аналитическая геометрия
Прогресс по курсу: 292/334

11 Собственные значения...

11.1 Основные понятия

11.2 Свойства собственн...

11.3 Нахождение собстве...

11.4 Математическая мод...

11.5 Интересные задачи

12 Квадратичные формы

12.1 Основные понятия

12.2 Знакоопределенные ...

12.3 Приведение квадрат...

12.4 Классификация крив...

13 Кривые второго порядка

Используя критерий Сильвестра, определить характер знакоопределенности квадратичной формы

$$f(x_1, x_2, x_3) = x_1^2 + 4x_1x_2 + 6x_1x_3 + 5x_2^2 + 10x_2x_3 + 15x_3^2.$$

Заполните пропуски

Вычислим угловые миноры:

$\Delta_1 =$

$\Delta_2 =$

$\Delta_3 =$

Ответ. Квадратичная форма является

Отправить

Выбрать:

Выбрать:

- положительно определенной.
- отрицательно определенной.
- неопределенной.

Рис. 5. Комбинирование режимов ввода ответа («Выбор» и «Текст») в одном задании

применяются, охватывая свыше 50 % практических задач — от базовых расчётов вероятности до анализа статистических данных.

Рассмотрим несколько примеров.

Задача 1. *Игральная кость брошена три раза. Найти вероятность того, что сумма выпавших очков равна 15.*

Реализация на Stepik. Событие A — сумма выпавших очков равна 15.

Шаг 1: Общее количество исходов

$$n = [\text{пропуск}].$$

Шаг 2: Число благоприятных исходов

$$m = [\text{пропуск}].$$

Шаг 3: Вероятность

$$P(A) = [\text{пропуск}].$$

Задача 2. *Дискретная случайная величина X может принимать только два значения:*

$$x_1 = 2 \text{ и } x_2 = 4.$$

Известна вероятность $p_1 = 0,4$. Вычислите математическое ожидание $M(X)$.

Реализация на Stepik.

Шаг 1: Вероятность $p_2 = [\text{пропуск}]$, так как $p_1 + p_2 = 1$.

Шаг 2: Математическое ожидание: $M(X) = 2 \cdot [\text{пропуск}] + 4 \cdot [\text{пропуск}] = [\text{пропуск}]$.

Задача 3. *Имеются две урны: в первой 3 белых шара и 5 чёрных; во второй 4 белых и 3 чёрных. Из первой урны во вторую перекладывают, не глядя, один шар. Затем из второй урны берут один шар. Найти вероятность того, что извлечённый шар окажется белым.*

Реализация на Stepik. При изучении темы «Формула полной вероятности» студенты часто испытывают трудности с самостоятельным определением гипотез. На начальном этапе эта проблема решается путём явного перечисления гипотез в условии задания, например,

H_1 : «Из первой урны во вторую переложен белый шар»;

H_2 : «Из первой урны во вторую переложен чёрный шар».

Такой подход упрощает понимание алгоритма и обеспечивает автоматическую проверку всех этапов решения — от промежуточных до итоговых (рис. 6). Далее уровень поддержки постепенно снижается, и от студентов требуется самостоятельная формулировка гипотез.

Теория вероятностей и математическая статистика
Прогресс по курсу: 126/367

- 3.2 Классическое опреде...
- 3.3 Статистическое опред...
- 3.4 Геометрическое опре...
- 3.5 Задачи
- 4 Основные теоремы тео...**
- 4.1 Действия над события...
- 4.2 Условная вероятность...
- 4.3 Вероятность суммы с...
- 4.4 Вероятность появлен...
- 4.5 Формула полной веро...
- 4.6 Формула Байеса
- 4.7 Задачи

Имеются две урны: в первой 3 **белых** шара и 5 **чёрных**; во второй 4 **белых** и 3 **чёрных**. Из первой урны во вторую перекладывают, не глядя, один шар. После этого из второй урны берут один шар. Найти вероятность того, что этот шар будет белым.

Решение. Событие A — {из второй урны извлечен белый шар}.

Формулируем гипотезы:

H_1 — {из первой урны во вторую переложен белый шар};

H_2 — {из первой урны во вторую переложен чёрный шар}.

Заполните пропуски

Вычислим соответствующие вероятности:

$P(H_1) =$ $P(A | H_1) =$

$P(H_2) =$ $P(A | H_2) =$

По формуле **полной вероятности**

$P(A) =$

Рис. 6. Решение задачи на формулу полной вероятности с явно заданными гипотезами

Задания формата «Пропуски» идеально подходят для изучения статистических методов, так как большинство статистических расчетов предполагают выполнение многоступенчатых процедур. Например, построение доверительного интервала включает следующие последовательные шаги: выбор уровня доверия, расчёт стандартного отклонения, определение критического значения и вычисление границ интервала. Пример такого задания показан на рис. 7.

Задания формата «Пропуски» эффективны и при проверке статистических гипотез (рис. 8). Проверка гипотезы осуществляется в несколько последовательных этапов: формулировка нулевой и альтернативной гипотез; выбор подходящего статистического критерия; расчет эмпирического значения выбранного критерия; сопоставление полученного значения с критическим уровнем; принятие обоснованного решения о принятии или отклонении гипотезы.

Использование заданий с пропусками позволяет студентам систематически контролировать правильность своих рассуждений и расчетов, разбивая решение сложной задачи на отдельные этапы. Это снижает вероятность возникновения ошибок и способствует

глубокому усвоению материала. Важно отметить, что задания с пропусками могут включать формулирование выводов на основании полученных данных, например, «с вероятностью 0,95 среднее значение генеральной совокупности лежит в интервале от [нижняя граница] до [верхняя граница]» или «на уровне значимости 5% [есть основания] / [нет оснований] отклонить нулевую гипотезу в пользу альтернативной».

Теория вероятностей и математическая статистика

Прогресс по курсу: 126/367

7 Выборочный метод

7.1 Генеральная совокупн...

7.2 Вариационный и стат...

7.3 Числовые характерис...

7.4 Точечная оценка пара...

7.5 Интервальная оценка ...

7.6 Необходимая численн...

7.7 Задачи

8 Проверка статистическ...

8.1 Статистические гипот...

8.2 Проверка гипотезы о...

По выборке (из нормально распределенной генеральной совокупности) объема $n = 16$, найдены выборочная средняя $\tilde{x} = 20,2$ и исправленная выборочная дисперсия $S^2 = 0,64$.

Оценить **неизвестное математическое ожидание** при помощи доверительного интервала с надежностью $\gamma = 0,95$.

Ответ округлите до *третьего десятичного знака* после запятой.

Заполните пропуски

Ответ.

$t =$ - округлите до второго знака после запятой

$\Delta =$

Доверительный интервал:

$< \bar{x} <$

Рис. 7. Задание на построение доверительного интервала в формате «Пропуски»

Теория вероятностей и математическая статистика

Прогресс по курсу: 126/367

8 Проверка статистическ...

8.1 Статистические гипот...

8.2 Проверка гипотезы о...

8.3 Проверка гипотезы о...

8.4 Проверка гипотезы о ...

8.5 Проверка гипотезы о ...

8.6 Проверка гипотезы о ...

8.7 Задачи

9 Корреляционно-регресс...

Из 50 человек, покупающих в магазине кофе, 20 человек выбирают сорт «Арабика». Проверьте на уровне значимости 0,05 гипотезу о том, что **половина** покупателей выбирает данный сорт против альтернативной гипотезы, что данный сорт выбирает **менее половины** покупателей.

Наблюдаемое значение округлите до *второго десятичного знака* после запятой.

Заполните пропуски

$H_0 : p =$ $w =$

$Z_{\text{набл}} =$ $Z_{\text{крит}} =$

Ответ. На уровне значимости 0,05 Выбрать: отвергнуть гипотезу H_0

Рис. 8. Проверка гипотезы о параметре биномиального распределения

Количество и типы пропусков можно регулировать, что позволяет варьировать сложность заданий, учитывая разный уровень подготовки студентов.

Рассмотренные примеры использования заданий с пропусками демонстрируют их основные преимущества:

- Сложные задачи разбиваются на логические этапы, облегчая усвоение материала.

- Пошаговое заполнение пропусков развивает навык правильного оформления решений.
- Возможность вставлять пояснения и формулы в текст заданий укрепляет связь теории с практикой.
- Уровень сложности заданий варьируется: от простейших (с подсказками) до продвинутых (самостоятельных).
- Универсальность формата, применимого в различных дисциплинах.

4. ОНЛАЙН-КУРСЫ В УЧЕБНОМ ПРОЦЕССЕ: ОПЫТ ВНЕДРЕНИЯ, РЕЗУЛЬТАТЫ И ПРАКТИЧЕСКИЕ РЕКОМЕНДАЦИИ

Онлайн-курсы «Линейная алгебра и аналитическая геометрия» и «Теория вероятностей и математическая статистика» начали активно применяться в учебном процессе Института управления, экономики и финансов Казанского федерального университета с февраля 2023 года. Для анализа эффективности методики были сравнены результаты зимней экзаменационной сессии 2024 года (после внедрения курсов) с результатами зимней сессии 2023 года (до внедрения курсов). На примере дисциплины «Теория вероятностей и математическая статистика» наблюдается положительная динамика.

4.1. Анализ академических результатов

Как видно из данных, представленных в таблице 1, после внедрения онлайн-курса средний балл по дисциплине вырос на 6 пунктов, что соответствует относительному увеличению на 9,5 %. Также наблюдается положительная динамика в распределении оценок: доля отличных оценок выросла на 5 %, а доля неудовлетворительных оценок сократилась на 4 %.

Таблица 1. Сравнение результатов экзаменационной сессии по дисциплине «Теория вероятностей и математическая статистика» до и после внедрения онлайн-курса

Показатель	Зимняя сессия 2023 года (до внедрения)	Зимняя сессия 2024 года (после внедрения)	Изменение
Количество студентов	98	116	+18
Средний балл (по 100-балльной шкале)	63	69	+6 (+9,5 %)
Доля оценок «5» (отлично)	10 %	15 %	+5 %
Доля оценок «2» (неудовлетворительно)	13 %	9 %	-4 %

Особенно заметный прирост успеваемости наблюдался в темах, традиционно считающихся сложными. Например, если ранее во время письменного экзамена до 70 % студентов даже не брались за решение задач по теме «Проверка статистических гипотез», то после введения онлайн-курса доля таких студентов снизилась до 30 %. Кроме того, значительно сократилось число учащихся, теряющих баллы из-за ошибок в оформлении решений, так как формат «Пропуски» приучил их к пошаговой записи решения. Учитывая, что задания формата «Пропуски» составили ядро практической части курсов и охватили более 50 % всех заданий, полученные результаты убедительно свидетельствуют об эффективности данного формата для повышения академической успеваемости.

4.2. Качественные результаты и обратная связь

1. Увеличение вовлеченности студентов в учебный процесс. Использование интерактивных заданий помогло повысить интерес студентов к предмету. По данным статистики Stepik (функция «Класс»), более 90 % студентов ИУЭФ КФУ, записавшихся на курс, активно выполняли задания. Многие студенты в ходе неформального опроса отметили, что разбор теоретического материала и решение задач в онлайн-среде сделали изучение математики более увлекательным и понятным. В результате значительно выросло количество активных участников учебного процесса: студенты стали охотнее выполнять домашние задания и чаще обсуждать изучаемый материал как в аудитории, так и в самом онлайн-курсе, используя встроенную систему комментариев (дискуссий) на платформе Stepik.

2. Выявление проблемных модулей. Постоянный мониторинг успеваемости и подробная статистика, предоставляемая платформой Stepik, позволили выявить разделы, вызывающие наибольшие трудности у студентов (например, «Собственные векторы и значения» в линейной алгебре или «Проверка гипотез» в статистике). На основе этих данных проводилась дополнительная работа по улучшению качества подачи материала и усилению соответствующих разделов курсов за счет добавления разъясняющих примеров и дополнительных практических заданий.

3. Оптимизация рабочего времени преподавателей. Электронная среда позволила освободить преподавателей от рутинной проверки однотипных заданий, перенаправив освободившееся время на индивидуальные консультации со студентами и обсуждение более сложных и значимых вопросов.

4.3. Практические рекомендации

На основе отзывов студентов и собственного опыта использования платформы сформулированы следующие рекомендации по использованию заданий с пропусками в онлайн-курсах:

- Дозированное количество пропусков. Оптимальным считается наличие 3-8 пропусков на одно задание. Избыточное количество пропусков на странице рассеивает внимание, вызывает утомление и снижает мотивацию студентов.
- Комбинация форматов заданий. За заданием с пропусками полезно размещать задачи другого типа, например, «Численная задача» или «Математическая задача», где требуется дать только окончательный ответ. Это поддерживает разнообразие и повышает вовлеченность студентов, а также готовит их к итоговому контролю, где не будет промежуточных подсказок.
- Чёткая формулировка инструкций по вводу правильных ответов. Предоставьте точные указания по оформлению ответов: правила округления чисел, знаки-разделители (запятая или точка), необходимость указания единиц измерения, отсутствие пробелов. Такая инструкция избавляет от двусмысленности и технических ошибок, не связанных с пониманием материала.

Соблюдение перечисленных рекомендаций позволит в полной мере раскрыть потенциал заданий с пропусками, превращая их в мощный инструмент повышения успеваемости и укрепления знаний студентов. Применение такого подхода не только оптимизирует учебный процесс, но и развивает у обучающихся жизненно важные навыки критического мышления, необходимые для успешной профессиональной деятельности.

5. ЗАКЛЮЧЕНИЕ

В данной статье был обобщен практический опыт разработки и применения заданий формата «Пропуски» в онлайн-курсах по математическим дисциплинам на образовательной платформе Stepiк. Цель работы — представить данный формат как эффективный инструмент для организации учебного процесса и оценить его влияние на успеваемость студентов — была достигнута.

Проведенное исследование позволяет сделать следующие выводы:

1. Задания формата «Пропуски» обладают значительным дидактическим потенциалом. Они позволяют разбивать сложные математические задачи на последовательные этапы, обеспечивая студенту мгновенную обратную связь и возможность самоконтроля на каждом шаге решения. Это способствует формированию прочных навыков и осознанному пониманию алгоритмов, а не просто угадыванию ответа.
2. Формат является гибким и универсальным. Комбинирование режимов ввода («Выбор» и «Текст») позволяет создавать задания разного уровня сложности — от тренировочных с подробными подсказками до итоговых, требующих полностью самостоятельного решения, что обеспечивает возможность индивидуального подхода к обучению.
3. Внедрение онлайн-курсов с заданиями «Пропуски» показало положительное влияние на академические результаты студентов. Как продемонстрировали данные сравнительного анализа, после внедрения методики средний балл успеваемости вырос на 9,5 %, увеличилась доля отличных оценок и сократилось количество неудовлетворительных результатов.
4. Методика эффективна для освоения тем, традиционно вызывающих наибольшие затруднения. Представленные в работе практические рекомендации могут быть полезны преподавателям высшей школы, разрабатывающим образовательный контент для онлайн- и гибридного форматов обучения.

Список литературы

1. Суцев С. С. Обоснование выбора цифровой платформы для создания онлайн-курса по изучению иностранного языка // Вестник МГПУ. Серия: Информатика и информатизация образования. 2022. № 2(60). С. 114–126. doi:10.25688/2072-9014.2022.60.2.11
2. Бачанцев И. В., Газейкина А. И., Долгов А. В. Выбор образовательной платформы для создания онлайн курсов по программированию // Актуальные вопросы преподавания математики, информатики и информационных технологий. 2020. № 5. С. 204–211.
3. Li Sh. A Study on the Validity of the Grammar Fill-in-the-blank Questions of the National English Paper for College Entrance Examination in 2020 // Region-Educational Research and Reviews. 2020. Vol. 2, № 4. P. 41–46. doi:10.32629/rerr.v2i4.203
4. Kleijn S., Pander Maat H., Sanders T. Cloze Testing for Comprehension Assessment: The HyTeC-Cloze // Language Testing. 2019. Vol. 36, № 4. P. 553–572. doi:10.1177/0265532219840382
5. Zyкова Т. В., Шершнева В. А., Вайнштейн Ю. В., et al. E-learning Courses in Mathematics in Higher Education // Perspectives of Science and Education. 2018. Vol. 4, № 34. P. 58–65.
6. Макжанова Я. В. Создание тестов по теме «Проверка статистических гипотез» в электронной образовательной среде // Вестник Московского Государственного Областного Университета. Серия: Педагогика. 2019. № 4. С. 111–121. doi:10.18384/2310-7219-2019-4-111-121
7. Быкова И. А. Поэтапное формирование умственных действий при обучении школьников программированию // Информатика в школе. 2023. № 2(181). С. 79–86. doi:10.32517/2221-1993-2023-22-2-79-86

8. He X., Zhang N. Comparison of Mistakes on Multiple-Choice Question and Fill-in-the-Blank Examinations: A Retrospective Analysis // Journal of Chiropractic Education. 2024. Vol. 38, № 2. P. 100–105. doi: 10.7899/jce-23-8
9. Varghese Sh. S. Analysis of Testing with Multiple Choice versus Fill in the Blank Questions: Outcome-based Observations in Dental Subjects // International Journal of Dentistry and Oral Science. 2021. P. 2020–2024. doi:10.19070/2377-8075-21000397
10. Онлайн-курс «Линейная алгебра и аналитическая геометрия» [Электронный ресурс]. URL: <https://stepik.org/140060> (дата обращения 16.07.2025).
11. Онлайн-курс «Теория вероятностей и математическая статистика» [Электронный ресурс]. URL: <https://stepik.org/182381> (дата обращения 16.07.2025).
12. Справочный центр Stepik [Электронный ресурс]. URL: <https://help.stepik.org/article/54796> (дата обращения 16.07.2025).
13. Степанов А. В. Языки разметки. Часть 2: Основные средства форматирования // Компьютерные инструменты в образовании. 2008. № 2. С. 12–19.

Поступила в редакцию 10.06.2025, окончательный вариант — 16.07.2025.

БагOUTДИНОВА Альфия Гизетдиновна, кандидат технических наук, доцент кафедры общей математики, Институт математики и механики им. Н.И. Лобачевского, Казанский федеральный университет, bagoutdinova@rambler.ru

Computer tools in education, 2025

№ 2: 59–72

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2025-2-59-72

Developing and Using Fill-in-the-Blank Tasks for Online Courses on the Stepik Platform

Bagoutdinova A. G.¹, Cand. Sc., Associate Professor, bagoutdinova@rambler.ru

¹Kazan Federal University, 35 Kremlevskaya Street, 420008, Kazan, Russia

Abstract

The article examines the experience of creating and using "Fill-in-the-blanks" tasks in the online courses "Linear Algebra and Analytical Geometry" and "Probability Theory and Mathematical Statistics." These courses were specifically developed as a supplement to in-person classes for students of the Institute of Management, Economics, and Finance at Kazan Federal University and are hosted on the Stepik educational platform. A significant portion of the practical tasks in these courses is presented in the "Fill-in-the-blanks" format, which is highly effective. The key feature is that a single test item can incorporate several questions interspersed with text. This format allows not only for assessing the student's final answer but also for tracking their entire reasoning process when solving a problem. The paper provides specific examples from the courses that illustrate the capabilities of fill-in-the-blanks tasks and offers recommendations for their effective use. Empirical data confirm the positive impact of the courses' implementation, including a 9.5% increase in the average score and a reduction in the proportion of unsatisfactory grades on topics that were previously challenging. This article will be useful for educators interested in developing

effective assignments to enhance the quality of education and student motivation in online or hybrid learning environments.

Keywords: "Fill-in-the-blanks" tasks, Stepik educational platform, online mathematics courses.

Citation: A. G. Bagoutdinova, "Developing and Using Fill-in-the-Blank Tasks for Online Courses on the Stepik Platform," *Computer tools in education*, no. 2, pp. 59–72, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-59-72

References

1. S. S. Sushev, "Choosing an e-learning platform to create an online foreign language course," *Vestnik Moskovskogo Gorodskogo Pedagogicheskogo Universiteta. Seriya: Informatika i informatizatsiya obrazovaniya*, no. 2(60), pp. 114–126, 2022 (in Russian); doi:10.25688/2072-9014.2022.60.2.11
2. I. V. Bachantsev, A. I. Gazykina, and A. V. Dolgov, "Selection of educational platform for creation of online programming courses," *Aktual'nye voprosy prepodavaniya matematiki, informatiki i informatsionnykh tekhnologiy*, no. 5, pp. 204–211, 2020 (in Russian).
3. S. Li, "A study on the validity of the grammar fill-in-the-blank questions of the national English paper for college entrance examination in 2020," *Region-Educational Research and Reviews*, vol. 2, no. 4, pp. 41–46, 2020; doi:10.32629/rerr.v2i4.203
4. S. Kleijn, H. Pander Maat, and T. Sanders, "Cloze testing for comprehension assessment: The HyTeC-Cloze," *Language Testing*, vol. 36, no. 4, pp. 553–572, 2019; doi:10.1177/0265532219840382
5. T. V. Zykova, V. A. Shershneva, Yu. V. Vainshtein et al., "E-learning courses in mathematics in higher education," *Perspectives of Science and Education*, no. 4(34), pp. 58–65, 2018.
6. Ya. V. Makzhanova, "Creating quizzes on the topic "statistical hypotheses testing" in e-learning environment," *Bulletin of the Moscow State Regional University (Pedagogics)*, no. 4, pp. 111–121, 2019 (in Russian); doi:10.18384/2310-7219-2019-4-111-121
7. I. A. Bykova, "Step-by-step formation of mental actions when teaching programming for schoolchildren," *Informatics in school*, no. 2(181), pp. 79–86, 2023 (in Russian); doi:10.32517/2221-1993-2023-22-2-79-86
8. X. He and N. Zhang, "Comparison of mistakes on multiple-choice question and fill-in-the-blank examinations: A retrospective analysis," *Journal of Chiropractic Education*, vol. 38, no. 2, pp. 100–105, 2024; doi:10.7899/jce-23-8
9. S. S. Varghese, "Analysis of testing with multiple choice versus fill in the blank questions: Outcome-based observations in dental subjects," *International Journal of Dentistry and Oral Science*, pp. 2020–2024, 2021; doi:10.19070/2377-8075-21000397
10. A. G. Bagoutdinova, "Linear algebra and analytic geometry. Online course," in [textistepik.org](https://stepik.org/140060), 2025 (in Russian). [Online]. Available: <https://stepik.org/140060>
11. A. G. Bagoutdinova, "Probability theory and mathematical statistics. Online course," in [textistepik.org](https://stepik.org/182381), 2025 (in Russian). [Online]. Available: <https://stepik.org/182381>
12. Stepik, "Stepik Help Center," in [textistepik.org](https://help.stepik.org/article/54796), 2025 (in Russian). [Online]. Available: <https://help.stepik.org/article/54796>
13. A. V. Stepanov, "Markup Languages. Part 2: Basic Formatting Tools," *Computer tools in education*, no. 2, pp. 12–19, 2008 (in Russian).

Received 10-06-2025, the final version — 16-07-2025.

Alfiya Bagoutdinova, Candidate of Sciences (Tech.), Associate Professor, Department of General Mathematics, Kazan Federal University, bagoutdinova@rambler.ru

АВТОМАТИЗАЦИЯ ВЫБОРА НАЧАЛЬНЫХ ПРИБЛИЖЕНИЙ ДЛЯ РЕШЕНИЯ НЕЛИНЕЙНЫХ УРАВНЕНИЙ ЧИСЛЕННЫМИ МЕТОДАМИ

Винокурова Д. В.¹, аспирант, ✉ d.v.vinokurova@gmail.com

¹Российский государственный педагогический университет им. А. И. Герцена,
набережная реки Мойки, д. 48, 191186, Санкт-Петербург, Россия

Аннотация

Выбор начальных приближений для нахождения корней нелинейных уравнений влияет на сходимость методов. В статье представлен алгоритм поиска начальных приближений для вычисления корней нелинейных уравнений для различных численных методов, который применяется для составления учебных задач. Реализация выполнена на языке программирования Python. В статье приводится псевдокод алгоритма. Рассматриваются возможности указанного алгоритма и вспомогательных функций, а также подробно описывается процесс работы программы. Представлены результаты сравнительного анализа количества итераций, необходимых для нахождения корней с применением алгоритма поиска начальных приближений и ручного метода подбора интервалов локализации корня. Предложенный подход продемонстрировал свою эффективность на 520 различных нелинейных уравнениях.

Ключевые слова: *начальные приближения, корни нелинейных уравнений, вычислительная математика, Python, численные методы*

Цитирование: Винокурова Д. В. Автоматизация выбора начальных приближений для решения нелинейных уравнений численными методами // Компьютерные инструменты в образовании. 2025. № 2. С. 73–80. doi:10.32603/2071-2340-2025-2-73-80

1. ВВЕДЕНИЕ

Нахождение корней нелинейных уравнений является важной задачей численного анализа. Существуют различные методы решения данных уравнений, такие как метод половинного деления, метод простой итерации, метод Ньютона, метод хорд и многие другие. Каждый метод имеет свои особенности. Правильный выбор начальной точки или интервала влияет на сходимость метода за определенное количество итераций.

В рассмотренной литературе [1–4] при описании различных методов не делается акцент на том, каким образом получают начальные приближения для численных методов. Приводимые примеры ограничиваются функциями, которые не содержат асимптоты и существенные ограничения на область допустимых значений [2, 4]. В образовательных целях обучающимся для применения рассмотренных численных методов к решению задач по нелинейным уравнениям преподавателю необходимо предоставлять различные типовые задачи и интервалы локализации корней для схождения методов за

определенное количество итераций при проведении расчетов вручную. Очевидно, что чем лучше будут выбраны интервалы, тем меньше итераций потребуется для сходимости выбранного численного метода. В связи с этим требуется автоматизация процесса поиска начальных приближений для дальнейшего вычисления корней нелинейных уравнений.

В работе предлагается алгоритм поиска начальных приближений для нахождения корней нелинейных уравнений на языке программирования Python. Предложенный подход сокращает количество итераций при поиске корней, учитывая области непрерывности и асимптоты функций. Приводится вспомогательный алгоритм для вычисления множества корней различными методами, опирающимися на функцию `root_scalar` из библиотеки SciPy [5].

2. АЛГОРИТМ ПОИСКА НАЧАЛЬНЫХ ПРИБЛИЖЕНИЙ

Для поиска начальных приближений у функций необходимо находить их области допустимых значений (ОДЗ), чтобы избежать отрезков, в которых имеются асимптоты.

Реализация алгоритма представлена в модуле `roots_nonlinear_eq` в функции `find_initial_approximations` [6]. Псевдокод для функции `find_initial_approximations` (ПОИСК_НАЧАЛЬНЫХ_ПРИБЛИЖЕНИЙ) представлен в Algorithm 1. Функция содержит один обязательный параметр `equation` (УРАВНЕНИЕ), который принимает на вход уравнение и четыре дополнительных параметра:

- `initial_range` (НАЧ_ИНТЕРВАЛ) — интервал для поиска начальных приближений, который задается в виде кортежа (x_{\min}, x_{\max}) , в псевдокоде (`X_МИН`, `X_МАКС`);
- `f_interval` (ИНТЕРВАЛ) — флаг наличия интервалов или отсутствия (для некоторых методов начальные приближения могут задаваться в виде одной точки либо в виде интервала);
- `num_points` (ЧИСЛО_ТОЧЕК) — количество точек для построения сетки;
- `ndigits` (КОЛИЧЕСТВО_ЗНАКОВ) — количество знаков после запятой (используется для компактного вывода учебных задач).

Алгоритм для функции `СОЗДАТЬ_ЗНАЧЕНИЯ_ПО_ОДЗ` приведен в Algorithm 2, данный код позволяет создать значения для x по ОДЗ функции.

Функция `СОЗДАТЬ_ДОПУСТИМЫЕ_ИНТЕРВАЛЫ` формирует список числовых интервалов заменяя бесконечности на конкретные числа, учитывая `НАЧ_ИНТЕРВАЛ`, заданный в функции `ПОИСК_НАЧАЛЬНЫХ_ПРИБЛИЖЕНИЙ`. Например, если интервал будет представлен в виде $(-\infty, 2)$, то при условии, что `X_МИН` = 1, `X_МАКС` = 3 функция вернет $[(1, 2)]$. Функция `ПОЛУЧИТЬ_ИСКЛЮЧЕННЫЕ_ТОЧКИ` извлекает точки, которые не должны входить в интервал (с учетом ОДЗ) и возвращает массив с исключенными точками.

Функция `СОЗДАТЬ_ЗНАЧЕНИЯ_ПО_ОДЗ` (`create_domain_values`) используется не для всех типов функций, в связи с тем, что:

- тригонометрические функции являются периодическими и в библиотеке SymPy их непрерывные области записываются через лямбда-функции. При наличии сложных функций внутри аргумента усложняется запись непрерывных интервалов и их необходимо раскрывать по определенным алгоритмам. Существует риск пропуска особенностей поведения функции, поэтому проводится проверка на наличие асимптот. Например, ОДЗ в SymPy в функции $\cot(5 \cdot x + 6) + (4 \cdot x + 9) / (2 \cdot x + 9)$ записывается в виде:

Algorithm 1:

ФУНКЦИЯ ПОИСК_НАЧАЛЬНЫХ_ПРИБЛИЖЕНИЙ(УРАВНЕНИЕ, НАЧ_ИНТЕРВАЛ = (-5, 5), ИНТЕРВАЛ = ИСТИНА, ЧИСЛО_ТОЧЕК = 1000, КОЛИЧЕСТВО_ЗНАКОВ = 6):

ОДЗ ← ПОЛУЧИТЬ_ОДЗ(УРАВНЕНИЕ, X)
 X_МИН ← НАЧ_ИНТЕРВАЛ[0]
 X_МАКС ← НАЧ_ИНТЕРВАЛ[1]

ЕСЛИ НЕ ЯВЛЯЕТСЯ_ТРИГОНОМЕТРИЧЕСКИМ(УРАВНЕНИЕ) **И**
НЕ (ЯВЛЯЕТСЯ_ДРОБНО_РАЦИОНАЛЬНЫМ(УРАВНЕНИЕ) **ИЛИ**
ЯВЛЯЕТСЯ_КОМБИНИРОВАННЫМ_ДРОБНЫМ(УРАВНЕНИЕ)) **И** ОДЗ $\neq \mathbb{R}$:

X_ЗНАЧЕНИЯ ← СОЗДАТЬ_ЗНАЧЕНИЯ_ПО_ОДЗ(ОДЗ,
 X_МИН, X_МАКС, ЧИСЛО_ТОЧЕК, КОЛИЧЕСТВО_ЗНАКОВ)

ИНАЧЕ:

X_ЗНАЧЕНИЯ ← РАВНОМЕРНАЯ_СЕТКА(X_МИН, X_МАКС, ЧИСЛО_ТОЧЕК)

Y_ЗНАЧЕНИЯ ← ПУСТОЙ_СПИСОК

ДЛЯ КАЖДОГО X_ТЕКУЩИЙ **ИЗ** X_ЗНАЧЕНИЯ:

ПОПРОБОВАТЬ:

Y_ТЕКУЩИЙ ← ВЫЧИСЛИТЬ(УРАВНЕНИЕ, X_ТЕКУЩИЙ)

ЕСЛИ ЯВЛЯЕТСЯ_ВЕЩЕСТВЕННЫМ(Y_ТЕКУЩИЙ):

ДОБАВИТЬ(Y_ЗНАЧЕНИЯ, Y_ТЕКУЩИЙ)

ИНАЧЕ:

ДОБАВИТЬ(Y_ЗНАЧЕНИЯ, NaN)

ЕСЛИ ОШИБКА:

ДОБАВИТЬ(Y_ЗНАЧЕНИЯ, NaN)

НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ ← ПУСТОЙ_СПИСОК

ДЛЯ i **ОТ** 0 **ДО** ДЛИНА(Y_ЗНАЧЕНИЯ) - 1:

Y1 ← Y_ЗНАЧЕНИЯ[i]

Y2 ← Y_ЗНАЧЕНИЯ[i + 1]

ЕСЛИ Y1 = NaN **ИЛИ** Y2 = NaN:

ПРОДОЛЖИТЬ

ЕСЛИ Y1 * Y2 < 0:

X1 ← ОКРУГЛИТЬ(X_ЗНАЧЕНИЯ[i], КОЛИЧЕСТВО_ЗНАКОВ)

X2 ← ОКРУГЛИТЬ(X_ЗНАЧЕНИЯ[i + 1], КОЛИЧЕСТВО_ЗНАКОВ)

ЕСЛИ НЕ ЯВЛЯЕТСЯ_ТРИГОНОМЕТРИЧЕСКИМ(УРАВНЕНИЕ) **И**
НЕ (ЯВЛЯЕТСЯ_ДРОБНО_РАЦИОНАЛЬНЫМ(УРАВНЕНИЕ) **ИЛИ**
ЯВЛЯЕТСЯ_КОМБИНИРОВАННЫМ_ДРОБНЫМ(УРАВНЕНИЕ)) **И** ОДЗ $\neq \mathbb{R}$:

ЕСЛИ ИНТЕРВАЛ:

ДОБАВИТЬ(НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, (X1, X2))

ИНАЧЕ:

ДОБАВИТЬ(НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, X1)

ИНАЧЕ ЕСЛИ НЕ ЯВЛЯЕТСЯ_ТРИГОНОМЕТРИЧЕСКИМ(УРАВНЕНИЕ):

ЕСЛИ ЗНАЧЕНИЯ_ВХОДЯТ_В_ОДЗ(X1, X2, ОДЗ, КОЛИЧЕСТВО_ЗНАКОВ=4):

ЕСЛИ ИНТЕРВАЛ:

ДОБАВИТЬ(НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, (X1, X2))

ИНАЧЕ:

ДОБАВИТЬ(НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, X1)

ИНАЧЕ:

ЕСЛИ АСИМПТОТЫ_НА_ИНТЕРВАЛЕ(УРАВНЕНИЕ, X, ЧИСЛО_ТОЧЕК, X1, X2):

ЕСЛИ ИНТЕРВАЛ:

ДОБАВИТЬ(НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, (X1, X2))

ИНАЧЕ:

ДОБАВИТЬ(НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, X1)

ВОЗВРАТ: НАЧАЛЬНЫЕ_ПРИБЛИЖЕНИЯ, X_ЗНАЧЕНИЯ, Y_ЗНАЧЕНИЯ

Algorithm 2:

ФУНКЦИЯ СОЗДАТЬ_ЗНАЧЕНИЯ_ПО_ОДЗ (
 ОДЗ,
 X_МИН,
 X_МАКС,
 ЧИСЛО_ТОЧЕК,
 КОЛИЧЕСТВО_ЗНАКОВ = 6):

ДОПУСТИМЫЕ_ИНТЕРВАЛЫ ← **СОЗДАТЬ_ДОПУСТИМЫЕ_ИНТЕРВАЛЫ**(ОДЗ, X_МИН, X_МАКС,
 КОЛИЧЕСТВО_ЗНАКОВ)
 ИСКЛЮЧЕННЫЕ_ТОЧКИ ← **ПОЛУЧИТЬ_ИСКЛЮЧЕННЫЕ_ТОЧКИ**(ОДЗ, КОЛИЧЕСТВО_ЗНАКОВ)

СПИСОК_СЕТОК ← ПУСТОЙ_СПИСОК
ДЛЯ КАЖДОГО (НАЧАЛО, КОНЕЦ) **ИЗ** ДОПУСТИМЫЕ_ИНТЕРВАЛЫ:
ДЛЯ КАЖДОЙ ТОЧКА **ИЗ** ИСКЛЮЧЕННЫЕ_ТОЧКИ:
 ЭПСИЛОН ← $10^{-КОЛИЧЕСТВО_ЗНАКОВ}$
ЕСЛИ ЧИСЛА ПОЧТИ РАВНЫ (КОНЕЦ, ТОЧКА, ЭПСИЛОН):
 КОНЕЦ ← КОНЕЦ - $10^{-КОЛИЧЕСТВО_ЗНАКОВ}$
ЕСЛИ ЧИСЛА ПОЧТИ РАВНЫ (НАЧАЛО, ТОЧКА, ЭПСИЛОН):
 НАЧАЛО ← НАЧАЛО + $10^{-КОЛИЧЕСТВО_ЗНАКОВ}$
ДОБАВИТЬ РАВНОМЕРНАЯ_СЕТКА (НАЧАЛО, КОНЕЦ, ЧИСЛО_ТОЧЕК / ЧИСЛО_ИНТЕРВАЛОВ,
 БЕЗ_ПОСЛЕДНЕЙ_ТОЧКИ) **В** СПИСОК_СЕТОК
ЕСЛИ ДЛИНА(ИСКЛЮЧЕННЫЕ_ТОЧКИ) = 0:
ДОБАВИТЬ РАВНОМЕРНАЯ_СЕТКА(НАЧАЛО, КОНЕЦ, ЧИСЛО_ТОЧЕК / ЧИСЛО_ИНТЕРВАЛОВ,
 БЕЗ_ПОСЛЕДНЕЙ_ТОЧКИ) **В** СПИСОК_СЕТОК
 ОБЩИЙ_МАССИВ ← **КОНКАТЕНАЦИЯ**(СПИСОК_СЕТОК)
ВОЗВРАТ: ОБЩИЙ_МАССИВ

$\text{Union}(\text{Complement}(\text{Interval.open}(-\infty, -9/2), \text{Union}(\text{ImageSet}(\text{Lambda}_n, 2^*n^*\pi/5 - 6/5 + 2^*\pi/5), \text{Integers}), \text{ImageSet}(\text{Lambda}_n, 2^*n^*\pi/5 - 6/5 + 3^*\pi/5), \text{Integers})), \text{Complement}(\text{Interval.open}(-9/2, \infty), \text{Union}(\text{ImageSet}(\text{Lambda}_n, 2^*n^*\pi/5 - 6/5 + 2^*\pi/5), \text{Integers}), \text{ImageSet}(\text{Lambda}_n, 2^*n^*\pi/5 - 6/5 + 3^*\pi/5), \text{Integers})))$;

- в дробно-рациональных уравнениях и комбинированных уравнениях, включающих комбинацию дробно-рациональных функций с другими типами функций при таком способе формирования значений, могут получаться лишние корни, из-за слишком близких значений вблизи асимптот.

Далее под особыми типами функций в тексте статьи будем подразумевать тригонометрические, дробно-рациональные и комбинированные функции, которые включают дробно-рациональные и другие типы подфункций. Классификация функций осуществляется программой при помощи специальных условий и соответствующих функций проверки.

Нахождение начальных приближений выполняется в цикле, в котором значения выбранного отрезка проверяются на значения NaN. При отсутствии данных значений во втором условии оцениваются концы отрезка на наличие различных знаков. Выполнение условия свидетельствует о существовании корня в данном интервале. Внутри условия содержатся три дополнительных: в первом условии проверяется, содержит ли данное уравнение особые типы функций и не включает ли непрерывная область всю область действительных чисел, во втором проверяется не относится ли функция к классу тригонометрических, третье условие выполняется в случае невыполнения первых двух.

В каждом из трех условий в начальное приближение добавляется интервал либо точка, в зависимости от флага ИНТЕРВАЛ, который задается исходя из применяемого метода.

Если функция не является тригонометрической (то есть выполняется второе условие), то дополнительно проверяется вхождение рассматриваемого интервала в непрерывную область в функции ЗНАЧЕНИЯ_ВХОДЯТ_В_ОДЗ. Если функция включает тригонометрическую функцию, то в начальном приближении проверяется наличие асимптот. Попадание в асимптоту может привести к бесконечным итерациям или лишним корням. Наличие асимптот проверяется в функции АСИМПТОТЫ_НА_ИНТЕРВАЛЕ, где интервал делится на заданное количество точек в зависимости от значения ЧИСЛО_ТОЧЕК. В каждой из этих точек выполняется проверка на асимптоты дополнительной функцией, в которой вычисляются пределы слева и справа, а затем проверяются на наличие резких скачков функции. В качестве резкого скачка функции установлено значение 50. Вернемся к условию в цикле формирования начальных приближений. В условии, в котором функция является тригонометрической, если в интервале отсутствует асимптота, в зависимости от флага ИНТЕРВАЛ, добавляется соответствующее приближение.

Функция ПОИСК_НАЧАЛЬНЫХ_ПРИБЛИЖЕНИЙ предоставляет несколько приближений, если они были найдены на интервале НАЧ_ИНТЕРВАЛ, что позволяет найти все корни. Если корней достаточно много, что, как правило, может возникать в тригонометрических функциях, которые содержат асимптоты, необходимо увеличить значение ЧИСЛО_ТОЧЕК, для устранения больших скачков значений в данных функциях.

3. РЕЗУЛЬТАТЫ ПРИМЕНЕНИЯ АЛГОРИТМА

Рассмотрим результаты применения функции `find_initial_approximations` на практике. Для рационального нахождения корней нелинейных уравнений в модуле `roots_nonlinear_eq` была разработана дополнительная функция для вызова численных методов с соответствующими параметрами. Функция `find_roots_with_root_scalar` вычисляет корни опираясь на алгоритмы, представленные в функции `root_scalar` модуля SciPy [5]. Предложенные функции позволяют найти все корни у нелинейных уравнений на заданных интервалах. В модуле `examples` представлены варианты вызова различных типов функций из модуля `roots_nonlinear_eq` [6] с использованием восьми методов, реализованных в `root_scalar`.

Опираясь на функцию `find_roots_with_root_scalar` для поиска корней, была проведена оценка возможностей функции `find_initial_approximations` на результатах сравнения количества итераций, необходимых для поиска корней с применением алгоритма поиска начальных приближений и ручного метода подбора интервалов локализации корня. Ручной метод основывается на выборе интервалов по эскизу графика функции. Для простоты анализа в случае наличия нескольких корней у функций рассматривался только один корень. Из семи типов уравнений (логарифмических, показательных, тригонометрических, дробно-рациональных, иррациональных, полиномиальных и комбинированных) для анализа было взято по одному уравнению и осуществлены вычисления для каждого из восьми методов [6].

В процессе анализа было выявлено, что использование функции `find_initial_approximations` намного эффективнее ручного поиска интервала локализации корня. Применяемая функция позволила сократить количество итераций и предоставила интервалы, которые привели к сходимости численных методов, в отличие от ручного подбора (рис. 1). Результаты анализа для других функций доступны в Zenodo в файле Excel [6]. Отметим, что в некоторых уравнениях количество итераций достигало 50. Это числовое ограничение по максимальному количеству итераций, которое могло иметь значительно большее

значение.

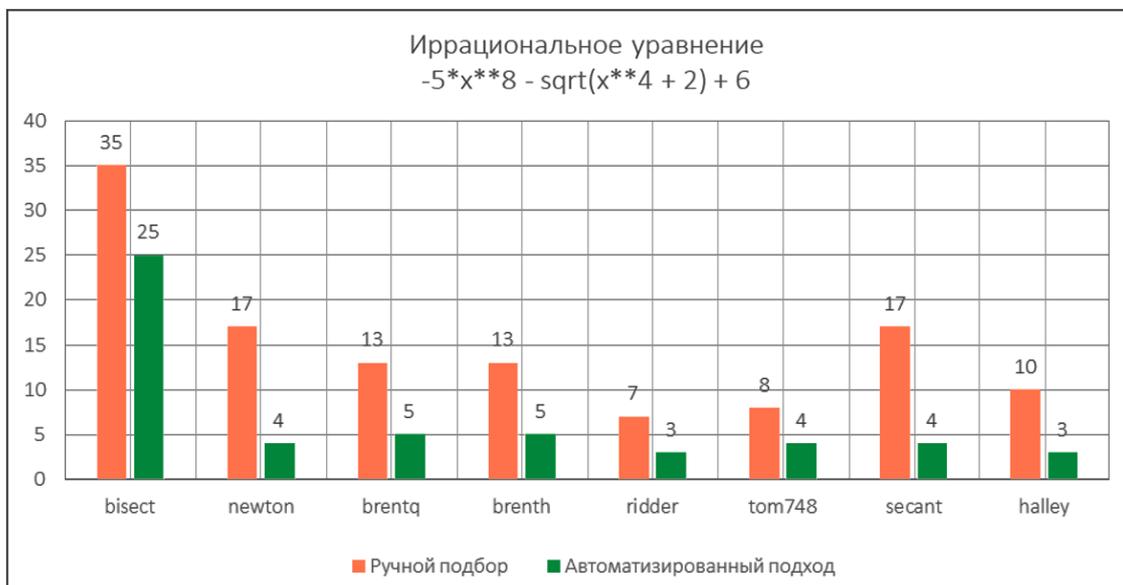


Рис. 1. Сравнительная гистограмма необходимого количества итераций для решения иррационального уравнения различными методами

В ходе функционального тестирования было установлено, что описанный метод поиска начальных приближений позволяет найти множество корней, существующих на выбранном интервале для различных типов функций. В некоторых задачах, в случае недостающих корней, возникает необходимость в увеличении значения ЧИСЛО_ТОЧЕК. Как правило, это происходит в уравнениях, содержащих функции с асимптотами.

При использовании предложенного метода с другими типами функций, выходящими за рамки протестированных случаев, рекомендуется экспериментальная проверка результатов для подтверждения корректности и устойчивости работы алгоритма.

4. ЗАКЛЮЧЕНИЕ

Алгоритм нахождения начальных приближений для нелинейных уравнений показал свою успешность на тестах семи различных типов уравнений, которые в общей сложности составили 520 уравнений, что подтверждает его корректность на использованном множестве функций. Сравнительный анализ количества итераций, необходимых для поиска корня каждым методом, показал, что предложенный подход значительно сокращает поиск корней и обеспечивает сходимость численных методов. Предложенный подход способствует эффективному и рациональному выполнению вычислений в области вычислительной математики. В образовательных целях данный алгоритм позволит обучающимся применять полученные начальные приближения для поиска корней нелинейных уравнений. Это обеспечит нахождение корней за минимальное число итераций. Программа реализована в виде модуля Python. Модуль размещён в открытом репозитории Zenodo [6], откуда его можно скачать и импортировать в собственные проекты.

Список литературы

1. *Зенков А. В.* Вычислительная математика для IT-специальностей : учебное пособие. М., Вологда: Инфра-Инженерия, 2022.
2. *Мицель А. А.* Вычислительные методы : учебное пособие. Томск: Эль Контент, 2013.
3. *Агапова Е. Г.* Вычислительная математика : учебное пособие. Хабаровск: Изд-во Тихоокеан. гос. ун-та, 2017.
4. *Коледин В. В.* Вычислительная математика : учебное пособие. Нижневартовск, 2023.
5. SciPy documentation. Version: 1.14.1. [Электронный ресурс]. URL: <https://docs.scipy.org/> (дата обращения: 15.11.24).
6. *Винокурова Д. В.* Программная реализация алгоритма поиска начальных приближений для решения нелинейных уравнений и анализ результатов. Zenodo, 2024. doi:10.5281/zenodo.17350251

Поступила в редакцию 10.05.2025, окончательный вариант — 16.07.2025.

Винокурова Дарья Валентиновна, аспирант, Институт информационных технологий и технологического образования, РГПУ им. А. И. Герцена, ✉ d.v.vinokurova@gmail.com

Computer tools in education, 2025

№ 2: 73–80

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2025-2-73-80

Automation of Initial Approximations Choice for Solving Nonlinear Equations by Numerical Methods

Vinokurova D. V.¹, Postgraduate, ✉ d.v.vinokurova@gmail.com

¹ Herzen University, 48 Moika river embankment, 191186, Saint Petersburg, Russia

Abstract

The choice of initial approximations for finding the roots of nonlinear equations affects the convergence of methods. This paper presents the algorithm for finding initial approximations for computing the roots of nonlinear equations for various numerical methods. This algorithm is used for compiling educational tasks. The implementation is written in the Python programming language. The article provides pseudocode for the algorithm. The possibilities of the specified algorithm and auxiliary functions are considered, and the process of the program operation is described in detail. The results of a comparative analysis of the number of iterations required to find roots using the initial approximation search algorithm and the manual method of selecting root localization intervals are presented. The proposed approach has demonstrated its efficiency on 520 different nonlinear equations.

Keywords: *initial approximations, roots of nonlinear equations, computational mathematics, Python, numerical methods.*

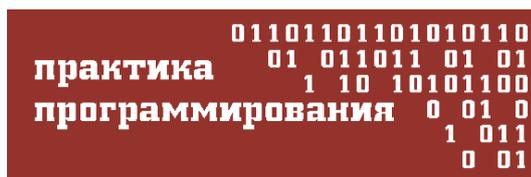
Citation: D. V. Vinokurova, "Automation of Initial Approximations Choice for Solving Nonlinear Equations by Numerical Methods," *Computer tools in education*, no. 2, pp. 73–80, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-73-80

References

1. A. V. Zenkov, *Vychislitel'naya matematika dlya IT-spetsial'nostey: uchebnoe posobie* [Computational Mathematics for IT Specialties: A Textbook], Moscow, Vologda, Russia: Infra-Engineering, 2022 (in Russian).
2. A. A. Mitsel, *Vychislitel'nye metody: uchebnoe posobie* [Computational Methods: A Textbook], Tomsk, Russia: El Content, 2013 (in Russian).
3. E. G. Agapova, *Vychislitel'naya matematika: uchebnoe posobie* [Computational Mathematics: A Textbook], Khabarovsk, Russia: Pacific State Institute Publishing House, 2017 (in Russian).
4. V. V. Koledin, *Vychislitel'naya matematika: uchebnoe posobie* [Computational Mathematics: A Textbook], Nizhnevartovsk, Russia, 2023 (in Russian).
5. SciPy Community, *SciPy documentation*, Version 1.14.1, 2024. [Online]. Available: <https://docs.scipy.org/>.
6. D. V. Vinokurova, "Programmnyaya realizatsiya algoritma poiska nachal'nyh priblizhenij dlya resheniya nelinejnyh uravnenij i analiz rezul'tatov" [Software Implementation of the Algorithm for Searching Initial Approximations for Solving Nonlinear Equations and Analysis of the Results], *Zenodo*, 2024 (in Russian). [Online]. Available: <https://zenodo.org/record/17350251>. doi:10.5281/zenodo.17350251

Received 10-05-2025, the final version — 16-07-2025.

Daria Vinokurova, Postgraduate, Institute of Computer Science and Technology Education, Herzen University, ✉ d.v.vinokurova@gmail.com



СОРЕВНОВАНИЯ ПО ПРОГРАММИРОВАНИЮ НА ПЛАТФОРМЕ ЯНДЕКС.КОНТЕСТ: КОМУ НУЖНЫ И КАК СОЗДАВАТЬ

Половикова О. Н.¹, канд. физ.-мат. наук, доцент, ✉ polovikovaol@yandex.ru,
orcid.org/0000-0002-9403-1195

Смолякова Л. Л.¹, старший преподаватель, knaus.larisa@gmail.com,
orcid.org/0000-0001-8547-0324

¹ Алтайский государственный университет, пр. Ленина, д. 61, 656049, Барнаул, Россия

Аннотация

В данном исследовании акцентируется внимание на необходимости получения студентами IT-направлений подготовки опыта решения практических задач. Активное участие студентов в спортивном программировании вне зависимости от начальных навыков способствует получению такого опыта. Представлен подход к созданию состязаний с использованием библиотеки *testlib* на бесплатной российской платформе Яндекс.Контест. Программные модули платформы обеспечивают основные этапы жизненного цикла соревнования от генерации тестов к задачам до подведения итогов. Предоставлена возможность реализовать логику работы своей системы оценивания, учитывая набранные студентами баллы и штрафы. Используя настройки заданий можно ограничивать ресурсы памяти и время выполнения программного кода участников. Специальный модуль также формирует отдельный набор требований к способу решения задач, что является важным условием для проведения соревнований в рамках учебного процесса. Создавая банк заданий для соревнования с учётом пройденного материала и изучаемых курсов, можно «вшить» проведение соревнований в учебный процесс, чередуя индивидуальные и командные этапы. Предлагаемый подход апробирован на практике: два раза в семестр студенты младших курсов одного из институтов Алтайского госуниверситета участвуют в состязаниях, тренируя свои навыки быстрого и безошибочного написания программного кода. Опыт проведения подобных турниров показал готовность студентов и преподавателей осваивать новые формы учебных занятий.

Ключевые слова: *соревнование по программированию, автоматизированная проверка кода, чекер, интерактор, постпроцессор, генерация тестов, штрафные очки, баллы за решение, монитор участников, банк заданий, тесты, программный код.*

Цитирование: Половикова О. Н., Смолякова Л. Л. Соревнования по программированию на платформе Яндекс.Контест: кому нужны и как создавать // Компьютерные инструменты в образовании. 2025. № 2. С. 81-95. doi: 10.32603/2071-2340-2025-2-81-95

1. ВВЕДЕНИЕ

Современные тенденции кадрового рынка требуют от выпускников вузов опыта решения практических задач в их профессиональной деятельности. Эти требования предъ-

являют работодатели, а формируются они средой, в которую попадают выпускники после трудоустройства. «Главная и наиболее часто звучащая претензия работодателей к выпускникам вузов сегодня — оторванность полученных знаний от практики» [1]. Федеральные государственные образовательные стандарты высшего образования по направлению «Информатика и вычислительная техника» определяют компетенции по программам бакалавриата, которые можно обеспечить в том случае, если студенты получили достаточный опыт самостоятельной и коллективной разработки программ [2]. Поэтому для вузов с IT-направлениями подготовки в ближайшей перспективе должна быть решена задача по созданию специальных условий для развития навыков программной разработки [3, 4].

Именно от опыта исполнителя зависит результативность всех этапов жизненного цикла прикладного проекта: от построения концепции до внедрения. Прикладные навыки формируются не только за счёт выполненного объёма подобных работ, но и спецификой и разнородностью решаемых задач. Нетривиальные (глубокие по сложности) решения, основанные на эффективных алгоритмах, невозможно построить, если использовать только заранее известные шаблоны. Для программной реализации таких задач нужны универсальные инструменты: системное мышление и навыки программной разработки с учётом некоторых ограничений (время на разработку, используемые программной ресурсы, методология и др.).

Одним из очевидных способов получения практического опыта разработки является участие в соревнованиях по программированию: вовлечение в сам процесс олимпиадного движения [5, с. 52]. Анализ исследований в данной предметной области [5–8] показал высокую ценность таких состязаний. От участников требуется максимальное погружение в предметную область, концентрация всех внутренних сил для достижения результата. В ходе работы формируется команда и выстраиваются взаимоотношения, оцениваются личностные качества: стрессоустойчивость, коммуникабельность, целеустремленность.

Олимпиадное программирование предполагает решение нетривиальных задач с учётом ограничений. «Основным принципом таких соревнований является решение заданных алгоритмических задач с ограничениями на объём потребляемой памяти и на время выполнения программы. При этом учитывается правильность решения задания, время написания этого решения и количество неудачных попыток сдачи решения» [8, с. 7].

Конечно, существует множество федеральных и международных соревнований подобного рода. Но следует учитывать тот факт, что эти соревнования требуют от участников серьёзной подготовки и слаженной командной работы, так как рассчитаны на «профессионалов», которые уже обладают уникальными навыками. Сложно найти и актуализировать состязания, которые рассчитаны на «новичков», делающих первые шаги как разработчики (программисты). В рамках учебного курса для нескольких групп (учебный поток) востребованы состязания «локального» уровня, которые будут соответствовать программе обучения: полученной базе теоретических знаний, рассмотренным языкам и средам программирования, освоенным инструментам для проектирования решений. Подобные мероприятия можно «вшить» в образовательную траекторию профильного курса, для того чтобы каждый студент мог приобрести ценный опыт участника, оценить свои возможности и получить дополнительные стимулы для своего развития.

В данной работе представлен подход к созданию соревнований по программированию с использованием готового решения - платформы Яндекс.Контест [9]. Соревнование рассматривается как система-конструктор, основные элементы которой формируются

через web-интерфейс, не требуя дополнительного программного обеспечения. На основе подхода построены состязания для студентов первого и второго курсов направления подготовки 09.04.03: Прикладная информатика Института математики и информационных технологий Алтайского госуниверситета. Периодичность их проведения — один раз в семестр за несколько недель до начала сессии. В течение двух недель предлагается решить несколько десятков задач для различных предметных областей. При этом необходимо использовать конкретные парадигмы программирования и определенные структуры данных. Вес каждой задачи в итоговой сумме баллов зависит от её сложности, от числа неудачных попыток решения и от количества положительных вердиктов, полученных ранее другими участниками. Личные вклады участников (набранные баллы) учитываются при выставлении зачётов по предмету.

Авторы исследования не ставят перед собой цели доказать уникальность выбранной платформы перед другими площадками с подобным функционалом. В данной статье сформированы практические рекомендации проведения состязаний по программированию с учётом ресурсов самой платформы и вспомогательных модулей, которые автоматизируют часть процессов по подготовке и проверке заданий. Материалы исследования позволят сделать выводы относительно применимости такого подхода к конструированию состязаний в конкретном вузе. Комментарии по настройке соревнования помогут организаторам избежать ошибок и сократить временные затраты на создание таких турниров также и на других платформах.

2. ПОЧЕМУ ВЫБРАНА ПЛАТФОРМА ЯНДЕКС.КОНТЕСТ?

Основываясь на выводах компании Rubrain.com [10], можно выделить следующие популярные платформы для проведения онлайн-состязаний по направлению «спортивное программирование»: CodeForces, Яндекс.Контест, AllCups, HackerRank. Каждая из этих систем обладает необходимым функционалом для тестирования кода и для поддержки своего сообщества (тренировки, курсы, тренажеры, образовательные раунды и т. д.). Например, AllCups — это площадка от компании Mail.ru Group для проведения крупных соревнований, таких как «Цифровой прорыв». Для сторонних организаций могут быть предоставлены ресурсы платформы, но для этого нужно обсудить условия сотрудничества (в том числе и финансовые) с компанией Mail.ru, согласовать концепцию соревнования, задачи и другие детали. Платформа HackerRank также позволяет создавать свои соревнования сторонним организациям и компаниям по платной подписке, предоставляя возможность использовать базу заданий платформы и/или самостоятельно формировать базу заданий.

CodeForces [11] — российская платформа для создания соревнований (мэшапов). Задачи для мэшапа можно загрузить свои (должны быть предварительно подготовлены в системе Polygon) или выбрать из архива Codeforces с использованием API. CodeForces развивается как тренировочная и учебная платформа, так как прошедшие соревнования можно использовать для самостоятельной подготовки. Эту возможность студенты могут использовать для развития навыков олимпиадного программирования и для налаживания эффективного взаимодействия внутри команды. Но существенным ограничением платформы является требование к пользователям для создания мэшапов: данный интерфейс становится доступен, если зафиксировано участие не менее чем в трех официальных рейтинговых раундах CodeForces [12].

Яндекс.Контест — платформа с полной поддержкой жизненного цикла соревнования (контеста). Правила для состязаний и задания готовят и размещают организаторы. Так

же как в системе Polygon [13], подготовка и настройка заданий основываются на библиотеке *testlib* [14]. В Яндекс.Контест есть открытые для зарегистрированных пользователей соревнования (без ограничения по рейтингу). По оценке Rubrain.com [10] большинство участников — начинающие программисты (джуны) или просто учащиеся.

Опыт организаторов олимпиад по программированию [15, 16] показывает, что платформа Яндекс.Контест обладает полноценным набором возможностей для подготовки, создания и проведения соревнований (контеста) с перспективой многократного использования, что важно для учебных заведений. Функционалом для построения конкурса может бесплатно воспользоваться любой зарегистрированный пользователь, при этом возможна коллективная работа по его созданию и настройке. Используемая система прав доступа позволяет управлять возможностями участников, оперируя учебными подгруппами, группами и потоками. Служба поддержки по запросу предоставляет готовые наборы задач разного уровня сложности, что бесспорно ускоряет построение соревнования и расширяет банк заданий.

3. ОСНОВНЫЕ ЭЛЕМЕНТЫ КОНТЕСТА. КАК ОНИ ВЗАИМОДЕЙСТВУЮТ?

Яндекс.Контест позволяет конструировать системы, элементы которых по необходимости можно корректировать и настраивать непосредственно во время проведения соревнований. У преподавателей есть возможности: вносить изменения банк заданий, изменять подсистему проверки, настраивать подсистему оценивания, анализировать и комментировать программный код участников, отслеживать активности.

Как и на большинстве платформ с поддержкой библиотеки *testlib*, работа с кодом в Яндекс.Контест основана на использовании вспомогательных подпрограмм: генератор, валидатор, чекер и постпроцессор. Аналогично платформе CodeForces Яндекс.Контест позволяет экспортировать задачи из системы Polygon. В данный сервис уже встроены средства автоматизации и самопроверки конкурса, что позволяет сократить ресурсозатраты на подготовку соревнований. На рисунке 1 представлена схема взаимодействия подпрограмм (модулей), отвечающих за ввод заданий и проверку решений участников.

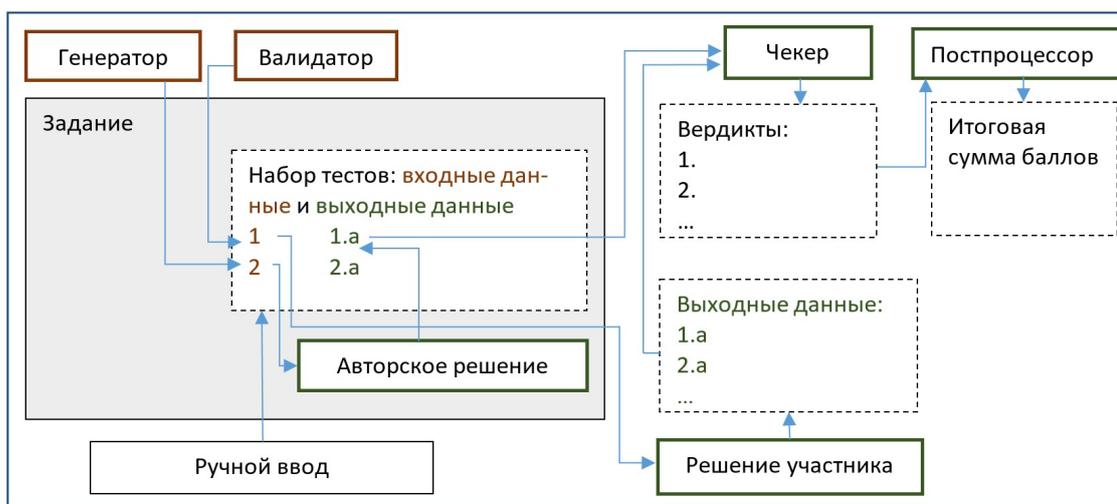


Рис. 1. Схема взаимодействия вспомогательных подпрограмм на платформе

Для автоматической проверки решений для каждого задания следует построить наборы тестов: варианты входных и выходных данных. Тесты можно создать *вручную* или воспользоваться генератором. Основная задача генератора — получить на выходе псевдослучайный набор данных, который и будет использоваться в качестве входных данных тестов. Авторское решение формирует выходные данные, которые система использует во время проверки пользовательских решений в качестве образцов.

Есть задачи, для которых сгенерировать возможные варианты входных данных крайне сложно, либо написание программы генерации является более затратным, чем *ручной ввод*. Для подобных задач нужен модуль — валидатор, который проверяет данные на соответствие некоторому шаблону, то есть отвечает за корректность самих тестов, на основе которых будет оцениваться программа участника. При ручном вводе тестов, валидатор позволяет избежать ошибок или неточностей, связанных, например, со знаками препинания или с форматами вещественных чисел.

Основная проверка решений осуществляется программой — чекером. Для обычных заданий чекер выполняет код участника на наборе тестов, сравнивает вывод программы участника с правильными ответами и сообщает системе результат сравнения в виде кодов возврата. Оценивать результат программы можно не только по символьному совпадению с шаблоном-ответа, но и руководствуясь специальными критериями, например, по принадлежности числового результата заданному диапазону, по включению полученного ответа в список, по соответствию сформированного текстового результата регулярному выражению.

Для специальных заданий, когда предъявляются, в том числе, требования к способу решения (например, использовать классы, рекурсию, функции и т. д.), к проверке подключается интерактор (см. рис. 2).

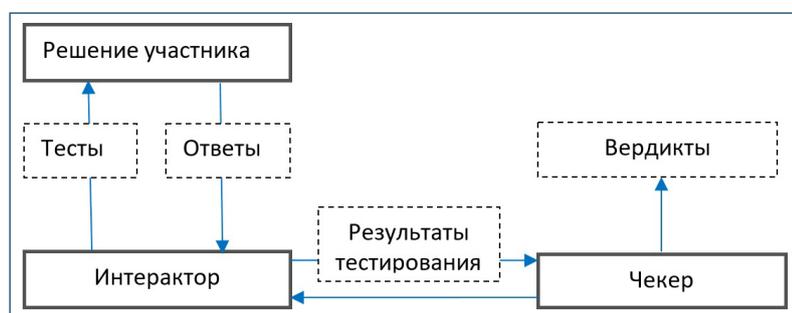


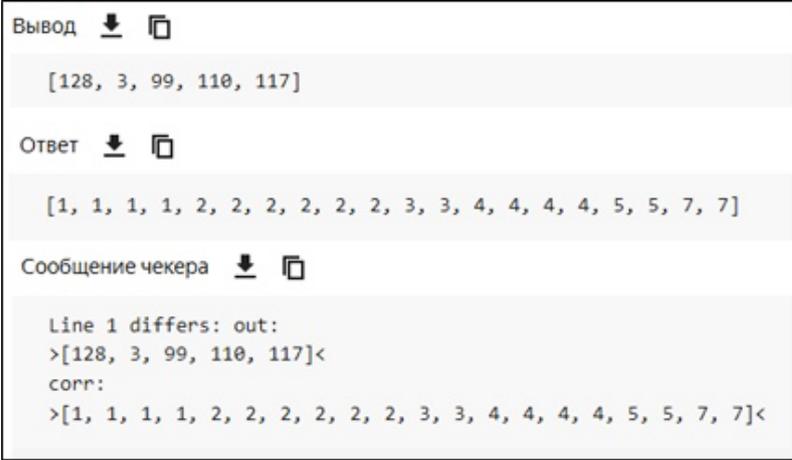
Рис. 2. Проверка решения с использованием интерактора

В этом случае для оценки программного кода выполняются следующие этапы: запускается интерактор, код динамически формирует тестовые данные для решения участника, проверяемая на корректность программа выполняется и полученный ответ анализируется интерактором. Таким образом, обеспечивается взаимодействие вида: *интерактор — проверяемый код — интерактор*. Количество необходимых запросов к коду может зависеть, например, от набора проверяемых функций программы или методов класса. Шаблоны (примеры) кода для создания интерактора можно загрузить с платформы Яндекс.Контест или из открытых репозиториях библиотеки *testlib*.

Для индивидуального оценивания каждой посылки участника нужно к заданию прописать логику работы постпроцессора. Чекер проверяет правильность работы программы на каждом тесте. Постпроцессор определяет итоговый балл за решение

и может управлять процессом оценивания. «Стоимость» решения в баллах может корректироваться, например в зависимости от количества пройденных тестов. Даже если проверяемая программа ошибается на некотором наборе тестов, постпроцессор может оценить это решение ненулевым баллом. «Штрафовать» участников можно за количество посылок с решением для каждой задачи, тем самым требуя от участников качественной разработки и тщательной предварительной проверки своего кода.

Также есть возможность открыть участникам доступ к отчетам чекера на проверяемых решениях. Студенты могут анализировать текстовые выводы (найденные ошибки) автоматической системы проверки решений (см. рис. 3), повторно отправлять исправленные варианты на проверку, отслеживать через специальный интерфейс (по веб-монитору) активность других участников, планировать и распределять свои усилия. Следует заметить, что данная платформа обеспечивает достаточный для учебного процесса выбор языков программирования, который отвечает современным тенденциям процесса разработки.



```
Вывод [128, 3, 99, 110, 117]
Ответ [1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 4, 4, 4, 4, 5, 5, 7, 7]
Сообщение чекера
Line 1 differs: out:
>[128, 3, 99, 110, 117]<
corr:
>[1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 3, 3, 4, 4, 4, 4, 5, 5, 7, 7]<
```

Рис. 3. Сообщение чекера по результатам проверки программного кода на тесте

Гибкость системы для проведения соревнования по программированию формируется не сложными многоступенчатыми настройками всех её параметров, а открытостью процессов работы с платформой. Можно запрограммировать свою логику работы всех подсистем соревнования, в рамках определенного набора правил и принципов.

4. С ЧЕГО НАЧАТЬ СОЗДАНИЕ СОРЕВНОВАНИЯ?

Для организации и проведения соревнований необходимо подготовить *Правила проведения и Положение о соревновании*, сформировать банк заданий, создать и настроить соревнование на специальной платформе, задать доступ для пользователей и определить их роли. *Правила* определяют цели, задачи и общие принципы в рамках которых будет строиться соревновательный процесс. *Положение* формируется с учётом *Правил*, определяет порядок и конкретные условия проведения соревнования по программированию: задания, сроки, этапы, участники, платформа и т. д. *Положение* формируется в процессе конструирования самого соревнования, потому что конкретные условия и ход выполнения его этапов зависят от тонкостей *программной настройки* контекста. Таким образом, в процессе работы будут получены конкретные правила проверки и оценивания реше-

<p>12. Распознать мебель на рисунке *</p> <p>Петя проходит стажировку в компании, которая продает мебель. Его попросили подписать фотографии мебели, чтобы их удобнее было добавлять в каталог. На каждой фотографии изображено несколько предметов мебели. Это могут быть шкафы, тумбочки или столы. Петя заметил, что вся мебель из темного дерева и сфотографирована на светлом фоне. Все шкафы - строго вертикальные прямоугольники, все столы - строго горизонтальные, а тумбочки - квадратные. Петя использовал стороннюю программу, чтобы получить черно-белое изображение, на котором остались только черные прямоугольники на белом фоне. Помогите Пете написать программу, которая облегчит его работу.</p> <p>Формат ввода</p> <p>В первой строке входных данных даются два числа n и m - размеры изображения. В следующих m строках вводится по n символов 0 (означает белый цвет) или 1 (черный). Причем все 1 в изображении образуют прямоугольники. Все прямоугольники расположены "внизу" изображения. Разные прямоугольники всегда разделены и НЕ находятся на изображении "вплотную" друг к другу.</p> <p>Формат вывода</p> <p>Для каждого прямоугольника на изображении слева направо вывести "ШКАФ", если прямоугольник изображен вертикально, "СТОЛ", если прямоугольник расположен горизонтально и "ТУМБОЧКА", если это квадрат.</p>	<p>12. Распознать мебель на рисунке *</p> <p>13. Супер код</p> <p>14. Поиск упаковок *</p> <p>15. Гадание на Рождество</p> <p>16. Палаточный лагерь *</p> <p>17. Выставка</p> <p>18. Теннис</p> <p>19. Подстрока в строке</p> <p>20. Подматрица из нулей</p> <p>21. Книги в библиотеке</p> <p>22. Градусник</p> <p>23. Англо-эльфийский словарь</p> <p>24. Самая большая цепочка</p>
---	--

Рис. 4. Пример задания (текст)

ний, процедуры формирования рейтинга участников и разрешения спорных ситуаций. В *Положении* следует обозначить и мотивационную составляющую соревнования, так как она во многом определяет его успешность. Это могут быть варианты возможных поощрений для студентов, которые преодолеют заданный в *Положении* порог баллов.

Так как соревнование должно соответствовать образовательному процессу, это в первую очередь накладывает ограничения на предлагаемый банк заданий: сложность заданий и стиль их текстового описания, ограничения на языки программирования и набор дополнительных требований по реализации. Примеры заданий и оценка их сложности представлены в работе Н. О. Котелина, Н. К. Попова [15]. Пример задания из соревнования по программированию для студентов Алтайского госуниверситета представлен на рисунке 4. Всего на данный момент в банке более сотни задач, решение которых предусматривает использование различных структур данных и алгоритмов поиска.

Количество тестов к заданиям определяется условиями конкретной задачи, но, как правило, для разносторонней проверки следует предусмотреть порядка 15–20 тестов. Один или несколько тестов (входные и выходные данные) можно добавить в качестве образца в примечание к задаче. Например, для задачи «Распознать мебель на рисунке» генератор формирует входные данные: два натуральных числа для размера изображения и бинарный массив, который и выступает изображением. Выходные данные (ответ) позволяют получить авторское решение. На рисунке 5 представлен пример такого примечания к задаче.

Для каждой задачи настраиваются индивидуальные требования к решению, включая ограничения на запуск, компиляцию и работу чекера. Требования контролируются автоматически в процессе проверки решения участника с формированием соответствующего вердикта. Закладываемые ограничения позволяют учитывать не только корректную работу алгоритма на некотором наборе тестов, но и время, ресурсы компиляции и выполнения кода, объём выводимых данных. Задачу можно включать в несколько соревнований, при этом совокупность требований и условий её выполнения может существенно отличаться. Тем самым можно значительно сэкономить ресурсы на создание новых за-

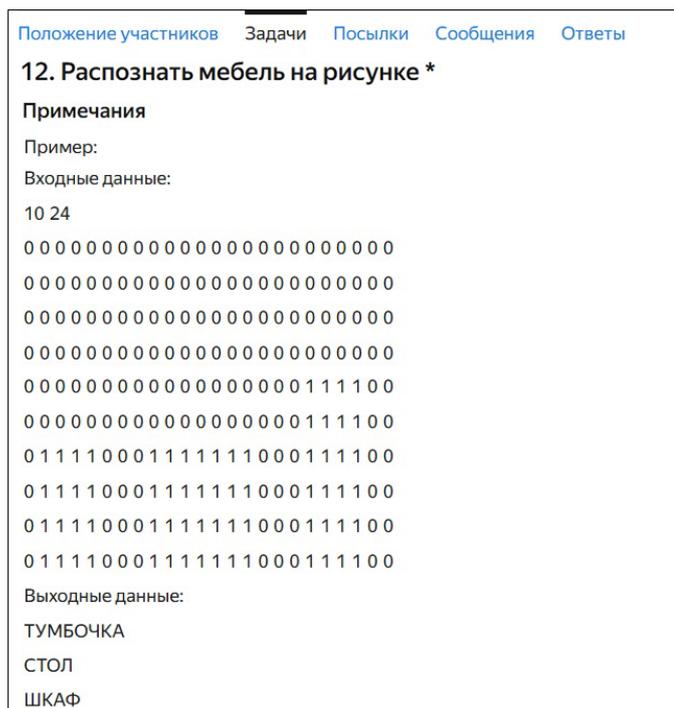


Рис. 5. Вариант теста к задаче в качестве примечания

дач. На едином банке заданий можно формировать соревнования различной сложности, ориентированные на подготовленность студентов.

5. КАК НАСТРОИТЬ ОЦЕНИВАНИЕ РЕШЕНИЙ?

В рамках рассматриваемого подхода, организаторы могут задать свою оценочную подсистему, которая каждому ответу участника (посылке) назначает баллы, таким образом рейтинг участника (его место в рейтинговой таблице) будет учитывать не только количество решённых задач. Расчет баллов и определение рейтинга также могут зависеть от других параметров, например, от количества неудачных посылок или потраченного времени [17]. В исследовании Г. О. Евстропова [18, с. 65] представлены базовые положения, на которые следует обратить внимание при построении оценочной подсистемы. На данный момент платформа Яндекс.Контест поддерживает 4 основных подхода к оцениванию участников, каждый из которых дополнительно настраивается в рамках соревнования:

SHAD — баллы за успешное решение рассчитываются как разность между «номинальной стоимостью» задачи и суммой штрафных баллов. Штрафные баллы можно назначить за неудачные посылки. Места в турнирной таблице определяются по убыванию итоговой суммы баллов.

АСМ — за каждую решённую задачу начисляется один балл. При равной сумме итоговых баллов при ранжировании участников учитываются минуты, потраченные на решение задач.

IOI — баллы за решение назначаются исходя из «номинальной стоимости» задачи, но могут быть переопределены логикой постпроцессора, исходя из количества пройденных

тестов. В дополнительных настройках данного подхода нет возможности настроить систему штрафов, но можно реализовать ее в постпроцессоре за «неудачные» тесты или в зависимости от потраченных минут. Ранжирование участников выполняется по убыванию итоговой суммы баллов.

SCORING — баллы за задачу выставляются чекером или постпроцессором. Ранжирование участников выполняется по убыванию итоговой суммы баллов, но при одинаковой сумме баллов можно учитывать штрафные очки (задаётся настройками).

Следует обратить внимание на подходы: IOI и SCORING, которые учитывают результаты работы постпроцессоров и позволяют настраивать правила штрафов за неудачные попытки. Гибкие настройки позволяют учесть уровень подготовки участников соревнований. Так, например, можно оценивать ненулевыми баллами решения, даже если не все тесты были пройдены (см. рис. 6). При этом можно поощрять дополнительными баллами решения, которые прошли проверку на всех тестах. Более опытных студентов можно штрафовать за потраченное время и за посылки с ошибками.

Следует стремиться к максимальной прозрачности при оценивании решений, заранее подготовить и публично представить правила проверки ответов. Студенты должны видеть работу оценочной подсистемы, которая начисляет или снимает баллы с «номинальной стоимости», опираясь на критерии. Так как соревнования по программированию встраиваются в учебный процесс подготовки выпускников ИТ-направлений, на занятиях учебных курсов можно разобрать работу дополнительных модулей (генератор,

Умеете программировать на Python

Игнорировать ошибки проверки кода перед компиляцией 🔴

Показывать пустые строки 🔴

Не игнорировать баллы чекера при непройденных тестах из условия 🔴

Видимость посылок во время заморозки Показывать число попыток ▾

Баллы за тест в задаче 0.05 баллов

Выбор оцениваемой посылки Лучшая по результату с токенами ▾

Задать баллы за тест в задаче 🔴

1	Очередь с приоритетом *	1 <input type="text"/> - <input type="text"/> + <input type="text"/>
2	Удаление из очереди *	2 <input type="text"/> - <input type="text"/> + <input type="text"/>
3	Угадай число	1 <input type="text"/> - <input type="text"/> + <input type="text"/>
4	Удаление повторяющихся символов	2 <input type="text"/> - <input type="text"/> + <input type="text"/>
5	Среднее квадратичное отклонение	1 <input type="text"/> - <input type="text"/> + <input type="text"/>
6	Медиана	1 <input type="text"/> - <input type="text"/> + <input type="text"/>
7	Расчет оценок учеников	1 <input type="text"/> - <input type="text"/> + <input type="text"/>

Рис. 6. Настройки контеста для оценивания решений участников

валидатор, постпроцессор, чекер, интерактор) и предложить студентам самостоятельно настроить их работу для конкретных задач. Шаблоны и примеры таких программ можно загрузить с официального сайта платформы.

Созданные студентами задачи можно использовать в качестве источника при обновлении существующего банка заданий. Знакомство с принципами работы дополнительных модулей в рамках создания контеста помогает студентам быстро интерпретировать и исправлять свои ошибки в коде, а также выстраивать тактику решения задач для получения максимального балла в соревнованиях.

6. УЧАСТНИКИ. ГДЕ И КАК СЛЕДИТЬ ЗА ХОДОМ СОРЕВНОВАНИЯ?

Участниками состязаний могут быть как зарегистрированные пользователи платформы, так и внешние пользователи. Расширенные возможности платформы по управлению ролями позволяют координировать работу пользовательских групп, которые можно создать и «связать» с учебными группами вуза. Следует также отметить проблему, с которой авторы статьи «столкнулись» при настройке групповой политики соревнования, — это ограничения по использованию кириллицы в названиях групп. Но несмотря на это неудобство, возможность оперировать учебными группами существенно сокращает затраты на работу с участниками соревнований. Платформа также обеспечивает решение всех технических вопросов по созданию и проведению командных соревнований. В этом случае итоговые баллы (и штрафы) суммируются, исходя из результатов всех участников команды.

Активности участников (набранные баллы, штрафные очки, количество посылок по задачам и т. д.) отображаются на специальном мониторе (см. рис. 7). Платформа в реальном времени генерирует данный интерфейс с актуальными результатами состязания. Тип монитора определяется выбранным подходом для системы оценивания. Интерфейс распределяет участников в порядке их рейтинга, учитывая итоговую сумму баллов и штрафы. Доступ к монитору настраивает администратор соревнования в зависимости

Соревнование по программированию на python для группы 4.305-1																			
Положение участников Задачи Посылки Сообщения Ответы																			
Последний правильный ответ: Задача 13 — toka.78 5м. 8с. Последнее отправленное решение: Задача 12 — toka.78 5м. 8с.																			
Показать реальный монитор																			
№ Участник	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	Очки	Штраф
	9/70	14/56	6/24	14/20	13/29	11/26	11/29	5/50	10/80	0/34	4/33	5/42	3/8	6/12	3/11	4/16	4/7		
1 student4206m3	+4 9д. 21ч.	+ 10д. 4ч.	+10 10д. 20ч.	+ 10д. 23ч.	+1 11д. 1ч.	+9 11д. 3ч.	+ 11д. 4ч.	+6 11д. 8ч.	+ 11д. 9ч.	-11 13д. 8ч.	+2 11д. 22ч.	+ 12д. 21ч.	+1 13д. 6ч.	+ 13д. 6ч.	+ 13д. 8ч.	+3 13д. 9ч.	+ 13д. 19ч.	40	952125
2 Convert.ToInt32	+3 00:16	+2 00:26	+ 00:31	+ 00:33	+2 00:48	—	+ 01:05	+2 01:10	+3 01:35	-5 02:07	+1 02:20	+15 03:32	+ 5д. 1ч.	+ 5д. 1ч.	+ 5д. 1ч.	+1 5д. 1ч.	+ 5д. 1ч.	20	74334
3 lva0412	+3 11д. 11ч.	+2 10д. 12ч.	—	+ 7д. 5ч.	+ 7д. 7ч.	+3 13:05	+4 7д. 1ч.	+3 12д. 9ч.	+8 8д. 6ч.	-11 4м. 7д.	+3 4м. 6д.	+2 9д. 13ч.	+1 4м. 9д.	+2 7д. 5ч.	—	+4 4м. 7д.	+1 4м. 7д.	16	1037522
4 dimasika	+8 2м. 1д.	+3 6д. 9ч.	+1 18д. 4ч.	+ 7д. 18ч.	+3 10д. 22ч.	+1 13д. 9ч.	+ 18д. 9ч.	+2 30д. 21ч.	+11 1м. 8д.	—	—	—	—	—	—	—	—	11	456755
5 sachacok	-2 1д. 2ч.	+2 01:07	—	+ 01:34	+1 1д. 1ч.	+ 1д. 1ч.	+ 2д. 19ч.	-10 2д. 19ч.	—	-1 2д. 19ч.	—	—	-3 1д.	—	—	—	—	9	26883
6 toka.78	—	+ 00:00	+ 00:00	+ 00:06	—	-1 03:38	—	—	+ 00:00	—	—	—	-1 5м. 8д.	—	—	—	—	8	46827
7 Viltaki	+3 1д.	+7 02:59	—	+ 04:03	+1 03:05	—	—	—	—	—	—	+22 3д. 21ч.	—	—	—	—	—	7	24035

Рис. 7. Монитор соревнования с промежуточным рейтингом участников

от роли пользователя. Опыт проведения соревнований показал: если актуальный монитор с посылками решений доступен не только для судей, но и для участников, можно наблюдать резкое увеличение («всплески») количества посылок к задачам, по которым кем-то из участников был получен успешный вердикт.

Такие «быстрые» решения, как правило, сырые и не проходят проверку тестированием. При этом участники тратят на эти решения время и силы. Чтобы избежать излишней эмоциональной нагрузки участников и переключить их внимание на продуктивную работу, в начале и несколько раз по ходу состязания необходимо «замораживать» монитор. Временная остановка монитора позволит участникам сосредоточиться на выборе подходов к решению и на проверке. Время и продолжительность заморозки можно регулировать в ходе проведения соревнования, опираясь на динамику изменения количества посылок к задачам.

Оперируя настройками соревнования, можно разрешить участникам продолжить выполнять задания, просматривать и сравнивать свои результаты после установленного времени завершения. Рейтинговая таблица фиксируется в момент завершения состязания (чтобы объявить результаты), но участники могут «дорешивать» предложенный набор задач, оставлять комментарии к решениям, задавать вопросы организаторам.

7. ЗАКЛЮЧЕНИЕ

С 2022 года спортивное программирование включено во Всероссийский реестр видов спорта. Данным исследованием авторы предприняли попытку обратить внимание на необходимость вовлечения студентов в такие состязания, которые тренируют навыки быстрого выбора алгоритма и скоростного безошибочного кодирования. Для успешного участия в соревнованиях по программированию участники должны иметь глубокие знания в различных научных дисциплинах, таких как топология, геометрия, дискретная математика, криптография и т. д. Это подталкивает студентов к самообразованию и к совершенствованию навыков командной работы с распределением ролей согласно сфере их интересов.

Анализ исследований [19, 20] показал прямую зависимость развития системного мышления у учащихся (студентов) от приобретения опыта решения практических задач. Этот вывод подтверждается тем, что, как отмечено в работе [21, с. 79], для перехода на уровень «описание системы» от учащихся требуется владение техниками системного мышления для глубокого анализа, либо соответствующий практический опыт. Именно такая практика развивает системное мышление, которое позволяет в будущем ставить и решать задачи из любой предметной области. Быстро разрабатывать эффективные по вычислительным ресурсам программы могут выпускники, которые на практике оценили разнообразие структур данных и базовых алгоритмов и приобрели опыт их адаптации под конкретную специфику предметной области.

Таким образом, можно сделать вывод, что опыт участия в подобных состязаниях является важным звеном всего образовательного процесса. Студенты учатся не только программировать в ограничениях, но и принимать решения в экстремальных условиях, анализировать работу модулей системы проверки, планировать порядок решения задач.

Представленный в статье подход к конструированию соревнований не требует дополнительных финансовых вложений и позволяет создавать состязания для студентов с разной начальной подготовкой. Перспективой развития данного подхода является подключение к подсистеме автоматизированной проверки решений дополнительного модуля —

детектора на «антиплагиат». Детектор, используя алгоритм «просеивания» метода «отпечатков», оценивает схожесть программных блоков для каждой новой посылки с уже принятыми решениями этой задачи [22]. Модуль не участвует в вынесении вердикта, но выявляет схожесть программ и помечает схожие фрагменты кода, информируя организаторов о возможной проблеме.

Список литературы

1. Раздел «Карьера» Санкт-Петербургского политехнического университета. Требования и претензии работодателей к выпускникам вуза [Электронный ресурс]. URL: <https://www.spbstu.ru/students/employment/demands-claims-employers-graduates/> (дата обращения: 10.07.2025).
2. Приказ Министерства образования и науки РФ от 19 сентября 2017 г. № 929 «Об утверждении федерального государственного образовательного стандарта высшего образования — бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника» (с изменениями и дополнениями) [Электронный ресурс]. URL: <http://publication.pravo.gov.ru/document/0001201710110018> (дата обращения: 10.07.2025).
3. Галахов Д. В. Формирование навыков командной работы и коммуникации у будущих специалистов в области информатики и вычислительной техники с использованием проектной методологии // Образование. Наука. Научные кадры. 2024. № 1. С. 333–342. doi:10.24412/2073-3305-2024-1-333-342
4. Селиванова И. В., Ураева Е. Е. Технологии и методы алгоритмического программирования в образовательном процессе при подготовке будущих it-специалистов Ученые записки // Электронный научный журнал Курского государственного университета. 2024. № 1. С. 267–273.
5. Крайванова В. А., Крючкова Е. Н. Олимпиадное программирование как эффективный инструмент подготовки профессиональных программистов // Вестник Новосибирского государственного университета: Информационные технологии. 2012. № 4. С. 51–56.
6. Горчаков Л. В., Стась А. Н., Карташов Д. В. Обучение программированию с использованием системы Ejudge // Вестник Томского государственного педагогического университета. 2017. № 9. С. 109–112.
7. Андреева Н. Л., Федорова А. Г. Саратовская школа программистов. Региональный опыт // Компьютерные инструменты в образовании. 2002. № 6. С. 51–55.
8. Алексеев А. В., Карелин В. А., Сеницын С. В. Студенческие соревнования по программированию в Югорском государственном университете // Вестник Югорского государственного университета. 2011. № 3. С. 7–9.
9. Платформа Яндекс.Контест [Электронный ресурс]. URL: <https://contest.yandex.ru/edu> (дата обращения: 10.07.2025).
10. Обзор платформ по проведению онлайн-чемпионатов. URL: <https://temofeev.ru/info/articles/obzor-platform-po-provedeniyu-onlayn-chempionatov/> (дата обращения: 10.07.2025).
11. Платформа CodeForces [Электронный ресурс]. URL: <https://codeforces.com/> (дата обращения: 10.07.2025).
12. Как создать мэшап на Codeforces [Электронный ресурс]. URL: <https://mksegment.ru/a/kak-sozdat-mehshap-na-codeforces> (дата обращения: 10.07.2025).
13. Платформа Poligon [Электронный ресурс]. URL: <https://polygon.codeforces.com/> (дата обращения: 10.07.2025).
14. Документация и исходные коды библиотеки testlib [Электронный ресурс]. URL: <https://github.com/MikeMirzayanov/testlib> (дата обращения: 10.07.2025).
15. Котелина Н. О., Попова Н. К. Подготовка интернет-тура чемпионата по программированию на Yandex.Contest // Вестник Сыктывкарского университета. 2018. № 26. С. 73–79.
16. Юрьев И. А., Гостева И. Н. Механизм и технология настройки онлайн-сервиса автоматизированной проверки заданий по информатике // Электронный научный журнал Курского государственного университета. 2023. № 2. С. 47–54.
17. Станкевич А. С. Общий подход к подведению итогов соревнований по программированию

при использовании различных систем оценки // Компьютерные инструменты в образовании. 2011. № 2. С. 27–38.

18. *Евстропов Г. О.* Системы оценивания в задачах с автоматической проверкой на олимпиадах по программированию // Информатика и образование. 2016. № 3. С. 65–67.
19. *Каримова Б. С.* Развитие системного мышления в опережающем обучении // Austrian Journal of Humanities and Social Sciences. 2015. № 9–10. С. 64–67.
20. *Галиев Т. Т., Ускенбаева Д. А.* Формирование системного мышления учащихся в процессе обучения // International scientific review. 2016. № 18. С. 92–93.
21. *Ковалев Г. О.* Системное мышление как компетенция // Вестник науки и образования. 2017. № 9. С. 72–79.
22. *Половикова О. Н., Иванова В. Е.* Разработка детектора автоматической проверки на плагиат блоков программного кода для образовательной среды // Высокопроизводительные вычислительные системы и технологии. 2020. № 1. С. 173–178.

Поступила в редакцию 05.03.2025, окончательный вариант — 10.07.2025.

Половикова Ольга Николаевна, кандидат физико-математических наук, доцент, доцент кафедры информатики, Алтайский государственный университет, ✉ polovikovaol@yandex.ru

Смолякова Лариса Ленгардовна, старший преподаватель кафедры информатики, Алтайский государственный университет, knaus.larisa@gmail.com

Computer tools in education, 2025

№ 2: 81–95

<http://cte.eltech.ru>

doi:10.32603/2071-2340-2025-2-81-95

Programming Competitions on the Yandex.Contest Platform: Who Needs Them and How to Create Them

Polovikova O. N.¹, Cand. Sc., Docent, ✉ polovikovaol@yandex.ru,
orcid.org/0000-0002-9403-1195

Smolyakova L. L.¹, Senior Lecturer, knaus.larisa@gmail.com, orcid.org/0000-0001-8547-0324

¹Altai State University, 61 Lenin Ave., 656049, Barnaul, Russia

Abstract

This study emphasizes the need for the IT students to gain experience in solving practical problems. Active participation in competitive programming, regardless of initial skills, helps students gain such experience. The article presents an approach to creating contests using Testlib library on the free domestic platform Yandex.Contest. The software modules cover the main stages of the competition life cycle, from generating tests for tasks to summarizing results. The use of a checker and postprocessor allows the implementation of the evaluation system logic, taking into account the points and penalties scored by the participants. The task configuration settings allow organizers to impose limits on memory usage and code execution time. The interactor program forms a separate set of requirements for the method of solving tasks, which is an important condition for holding competitions within the educational process. By creating a bank of tasks for competitions,

taking into account the material covered and curriculum, it is possible to «integrate» competitions into the educational process, alternating between individual and team stages. The proposed approach has been tested in practice: twice a semester junior students from one of the institutes of Altai State University participate in a competition, honing their skills in writing program code quickly and accurately. Experience with such competitions has shown that students and teachers are willing to embrace new forms of learning.

Keywords: *programming competition, automated code review, checker, interactor, postprocessor, test generation, penalty points, solution points, participant monitor, task bank, tests, program code.*

Citation: O. N. Polovikova and L. L. Smolyakova, "Programming Competitions on the Yandex.Contest Platform: Who Needs Them and How to Create Them," *Computer tools in education*, no. 2, pp. 81–95, 2025 (in Russian); doi:10.32603/2071-2340-2025-2-81-95

References

1. "Requirements and claims of employers to university graduates," in *www.spbstu.ru*, 2025. [Online] (in Russian). Available: <https://www.spbstu.ru/students/employment/demands-claims-employers-graduates/>
2. Ministry of Education and Science of the Russian Federation, "On Approval of the Federal State Educational Standard of Higher Education - Bachelor's Degree in the Field of Study 09.03.01 Informatics and Computer Engineering (as amended and supplemented)," *Order No. 929, Sep. 19, 2017*, 2017. [Online] (in Russian). Available: <http://publication.pravo.gov.ru/document/0001201710110018>
3. D. V. Galakhov, "Formation of teamwork and communication skills among future specialists in the field of it using project methodology," *Obrazovaniye. Nauka. Nauchnyye kadry = Education. Science. Scientific personnel*, no. 1, pp. 333–342, 2024 (in Russian); doi:10.24412/2073-3305-2024-1-333-342
4. I. V. Selivanova and E. E. Uraeva, "Technologies and methods of algorithmic programming in the educational process during preparation of future IT specialists," *Uchenye zapiski: Elektronnyi nauchnyi zhurnal Kurskogo gosudarstvennogo universiteta*, no. 1, pp. 267–273, 2024 (in Russian).
5. V. A. Kraivanova and E. N. Kryuchkova, "Olympiad programming as an effective tool for the training of professional programmers," *Vestnik NSU. Series: Information Technologies*, no. 4, pp. 51–56, 2012 (in Russian).
6. L. V. Gorchakov, A. N. Stas, and D. V. Kartashov, "Programming training using the ejudge system," *Tomsk state pedagogical university bulletin*, no. 9, pp. 109–112, 2017 (in Russian); doi:10.23951/1609-624x-2017-9-109-112
7. N. L. Andreeva and A. G. Fedorova, "Saratov school of programmers. Regional experience," *Computer tools in education*, no. 6, pp. 51–55, 2002 (in Russian).
8. A. V. Alekseev, V. A. Karelin, and S. V. Sinitsyn, "Student programming contest at Yugra State University," *Yugra State University Bulletin*, no. 3, pp. 7–9, 2011 (in Russian).
9. Yandex, *Yandex.Contest platform*, 2025. [Online] (in Russian). Available: <https://contest.yandex.ru/edu>
10. Y. Temofeev, "Overview of platforms for conducting online championships," *Temofeev.ru: IT, business, personal growth*, 2025. [Online] (in Russian). Available: <https://temofeev.ru/info/articles/obzor-platform-po-provedeniyu-onlayn-chempionatov/>
11. Codeforces, *CodeForces platform*, 2025. [Online] (in Russian). Available: <https://codeforces.com/>
12. Mksegment, "How to create a mashup on Codeforces," *MK Segment: Programming and Development*, 2025. [Online] (in Russian). Available: <https://mksegment.ru/a/kak-sozdat-mehshap-na-codeforces>
13. Polygon, *Polygon platform*, 2025. [Online]. Available: <https://polygon.codeforces.com/>
14. M. Mirzayanov, *Documentation and source code of the testlib library*, 2025. [Online]. Available: <https://github.com/MikeMirzayanov/testlib>
15. N. O. Kotelina and N. K. Popova, "The preparation of the online round of the championship on programming on Yandex.Contest platform," *Bulletin of Syktyvkar University Series 1: Mathematics. Mechanics. Informatics*, no. 26, pp. 73–79, 2018 (in Russian).
16. I. A. Yuriev and I. N. Gosteva, "Mechanism and technology of setting up an online service for automated verification of computer science assignments," *Elektronnyi nauchnyi zhurnal Kurskogo gosudarstvennogo universiteta*, no. 2, pp. 47–54, 2023 (in Russian).

17. A. S. Stankevich, "General approach to summarizing the results of programming competitions when using different scoring systems," *Computer tools in education*, no. 2, pp. 27–38, 2011 (in Russian).
18. G. O. Evstropov, "Scoring systems used for automatic evaluation in programming competitions in informatics," *Informatics and education*, no. 3, pp. 65–67, 2016 (in Russian).
19. B. S. Karimova, "Development of systems thinking in advanced training," *Austrian Journal of Humanities and Social Sciences*, no. 9-10, pp. 64–67, 2015.
20. T. T. Galiev and D. A. Uskenbaeva, "The formation of the system thinking of pupils in the learning process," *International scientific review*, no. 18, pp. 92–93, 2016 (in Russian).
21. G. O. Kovalev, "Systems thinking as a competence," *Vestnik nauki i obrazovaniia*, no. 9, pp. 72–79, 2017 (in Russian).
22. O. N. Polovikova and V. E. Ivanova, "Development of a detector of automatic check for plagiarized blocks of program code for educational environment," *Vysokoproizvoditelnye vychislitelnye sistemy i tekhnologii*, no. 1, pp. 173–178, 2020 (in Russian).

Received 05-03-2025, the final version — 10-07-2025.

Olga Polovikova, Candidate of Sciences (Phys.-Math.), Docent, Altai State University,
✉ polovikovaol@yandex.ru

Larisa Smolyakova, Senior Lecturer, Department of Informatics, Altai State University,
knaus.larisa@gmail.com