



А.А. Рахимов, С.К. Исмоилова

Политехнический институт Таджикского технического университета
имени академика М.С. Осими

ИСПОЛЬЗОВАНИЕ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ ПРИ РЕШЕНИИ СИСТЕМ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ МЕТОДОМ КРАМЕРА

В настоящей статье рассматривается отчет о разработке программы для решения систем линейных уравнений методом Крамера.

В рамках данной работы была разработана программа для решения систем линейных уравнений методом Крамера с помощью языка программирования Python. Этот метод используется в линейной алгебре для нахождения решений квадратных систем уравнений, при условии, что определитель матрицы коэффициентов не равен нулю. Программа реализована на языке Python с использованием библиотеки PyQt6, которая предоставляет инструменты для создания графического пользовательского интерфейса. Это позволяет пользователям легко вводить данные и получать результаты без необходимости взаимодействия с кодом.

Метод Крамера, Python, компьютерное моделирование, система линейных уравнений, графический интерфейс.

Введение

Компьютерные системы, программы и технологии, которые они позволяют использовать, играют все более важную роль в работе и повседневной жизни. Таким образом, способность использовать компьютеры для решения задач является важной компетенцией, которую студенты должны развивать, чтобы преуспеть в современном цифровом мире. Даже люди, которые не планируют карьеру в области вычислительной техники, могут получить пользу от развития навыков решения вычислительных задач, поскольку эти навыки улучшают понимание и решение широкого спектра проблем, выходящих за рамки информатики.

«Решения систем линейных уравнений методом Крамера» – это итеративный процесс разработки вычислительных решений проблем, которые выражаются в виде логических последовательностей шагов (т.е. алгоритмов), где каждый шаг точно определен, чтобы его можно было выразить в форме, которую может выполнить компьютер [3].

Известны имена российских ученых – М.А. Науменко (о преподавании математики с помощью технических средств и использования компьютерного моделирования), Т.А. Иванова, Д.Л. Егоренков, О.А. Тарасова, Э.Т. Селиванова [16], К.А. Федулова, Ж.И. Солодовиченко, Штофф В.А., Л.А. Шкутина и Т.В. Чернякова. В их работах обоснованы методы обучения математике посредством компьютерного моделирования, использования технологических и информационных средств, компьютерных программ при обучении геометрии, математических моделей в обучении будущих инженеров.

В работах современных таджикских исследователей С.Г. Гуломнабиева [5], З.А. Аминовой [8], А.А. Умарова [4], М. Нугмонова [6], Н.С. Абдуллоева [9] и А.А. Рахимова [1–3, 14] рассмотрены вопросы использования компьютерных программ, применения математического и компьютерного моделирования, а также использование технических средств в процессе обучения.

Следует отметить, что в настоящее время описано использование компьютерных программ в процессе обучения математике [15], использование языка программирования Python на теоретических и практических занятиях, что позволяет эффективно внедрять компьютерное моделирование в процессе обучения студентов технических вузов математике.

Постановка задачи

Постановка задачи заключается в разработке программного и системного обеспечения для решения задач вычислительного метода с использованием компьютерных программ и технологий. Целью разработки данной программы является создание удобного и эффективного инструмента для решения систем линейных уравнений. Основные задачи включают:

1. Реализация алгоритма метода Крамера для вычисления решений систем линейных уравнений.
2. Разработка графического интерфейса, который упрощает ввод данных и взаимодействие с пользователем.
3. Обработка возможных ошибок и исключений, что повышает надежность работы программы.
4. Обеспечение ясной визуализации результатов, чтобы пользователи могли легко интерпретировать полученные данные.

Новизна и актуальность научной работы

Создание программного обеспечения для решения систем линейных уравнений методом Крамера представляет собой инновационный подход, который будет вносить существенный вклад в образовательное пространство. Разработка таких программ обеспечит студентов и начинающих специалистов необходимыми инструментами для эффективного освоения и применения численных методов в обучении математике. Более того, данное программное решение будет способствовать развитию вычислительного мышления и повышению уровня компетенций в области программирования и математики, а также эти программы будут полезными и свое-

временными для преподавателей математики вузов для проверки письменных работ студентов.

Материалы и методы

В данной исследовательской работе будут использованы различные методы, такие как описание кодов, алгоритм программы и блок-схема. Это включает анализ существующих исследований, проведение экспериментов для тестирования программного обеспечения, проектирование интерфейса и модулей. Комбинация этих методов позволит получить полное понимание разрабатываемого программного решения и оценить его эффективность.

Основная часть

Описание программы.

Программа состоит из двух основных классов:

1. CramerSolver

Этот класс отвечает за математическую часть программы. Он принимает матрицу коэффициентов и вектор свободных членов, вычисляет определитель матрицы и решает систему уравнений методом Крамера.

Методы:

- `__init__(self, matrix, constants)`: конструктор, который принимает матрицу коэффициентов и вектор свободных членов и инициализирует соответствующие атрибуты;

- `determinant(self, mat)`: вычисляет определитель переданной матрицы с использованием функции `numpy.linalg.det()`. Определитель является ключевым элементом для применения метода Крамера;

- `solve(self)`: выполняет вычисления по методу Крамера. Этот метод сначала вычисляет определитель основной матрицы, а затем для каждой переменной заменяет соответствующий столбец на вектор свободных членов, чтобы найти ее значение. Если определитель равен нулю, выбрасывается исключение. Пример кода:

```
class CramerSolver:
```

```
    def __init__(self, matrix, constants):
        self.matrix = np.array(matrix, dtype=float)
        self.constants = np.array(constants, dtype=float)
        self.n = len(self.constants)
```

```
    def determinant(self, mat):
        return np.linalg.det(mat)
```

```
    def solve(self):
        det_main = self.determinant(self.matrix)
```

```
        if det_main == 0:
            raise ValueError("Система не имеет уникального решения (определитель основной матрицы равен нулю).")
```

```
        s_luti_ns = []
```

```
        for i in range(self.n):
```

```
            m_dified_matrix = self.matrix.copy()
```

```
            m_dified_matrix[:, i] = self.constants
```

```
            det_m_dified
```

```
self.determinant(m_dified_matrix)
```

```
            s_luti_ns.append(det_m_dified / det_main)
```

```
return s_luti_ns
```

2. CramerApp

Этот класс отвечает за графический интерфейс приложения. Он позволяет пользователю вводить размер матрицы, значения матрицы и вектора констант.

Методы:

- `__init__(self)`: конструктор, который инициализирует графический интерфейс приложения, включая создание необходимых элементов управления;

- `initUI(self)`: настраивает основной пользовательский интерфейс, включая поля ввода для размера матрицы, кнопки и сетку для ввода значений;

- `create_matrix_input_fields(self)`: создает поля ввода для матрицы и вектора констант на основе указанного размера. Этот метод динамически изменяет интерфейс, чтобы он соответствовал введенному пользователем размеру матрицы;

- `solve_system(self)`: считывает данные из полей ввода, создает объект `CramerSolver` и отображает результаты в диалоговом окне. Если данные неверны, будет выведено сообщение об ошибке.

Пример кода:

```
class CramerApp(QWidget):
```

```
    def __init__(self):
        super().__init__()
        self.initUI()
```

```
    def initUI(self):
```

```
        self.setWindowTitle('Cramer Solver')
```

```
        self.setGeometry(100, 100, 500, 400)
```

```
        layout = QVBoxLayout()
```

```
        self.size_label = QLabel('Размер матрицы (n x n):', self)
```

```
        layout.addWidget(self.size_label)
```

```
        self.size_input = QSpinBox(self)
```

```
        self.size_input.setMinimum(2)
```

```
        self.size_input.setMaximum(10)
```

```
        layout.addWidget(self.size_input)
```

```
        set_size_button = QPushButton('Установить размер', self)
```

```
        set_size_button.clicked.connect(self.create_matrix_input_fields)
```

```
        layout.addWidget(set_size_button)
```

```
        self.grid_layout = QGridLayout()
```

```
        layout.addLayout(self.grid_layout)
```

```
        self.solve_button = QPushButton('Решить', self)
```

```
        self.solve_button.clicked.connect(self.solve_system)
```

```
        self.solve_button.setEnabled(False)
```

```
        layout.addWidget(self.solve_button)
```

```
        self.setLayout(layout)
```

```
        self.apply_styles()
```

Как использовать программу

1. Запуск программы:

- Для запуска программы требуется установить Python и необходимые библиотеки, такие как **PyQt6** и **numpy**. Программа может быть запущена из терминала или с помощью любой интегрированной среды разработки (**IDE**).

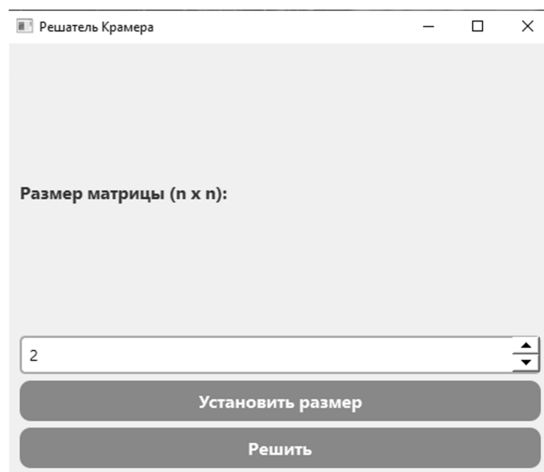


Рис. 1. Окно программы

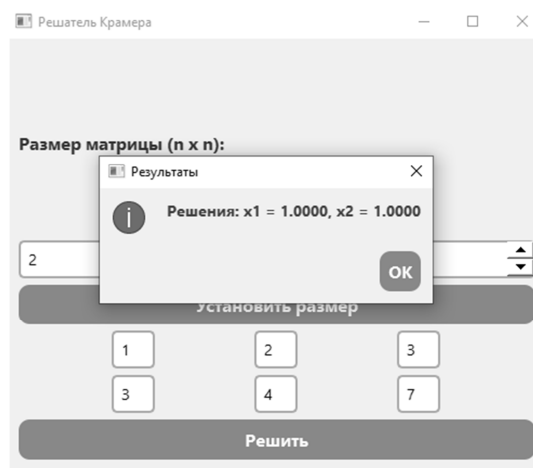


Рис. 2. Окно с полем для ввода размера матрицы

2. Ввод данных:

- После запуска программы пользователь видит окно с полем для ввода размера матрицы.
- Введите размер матрицы (от 2 до 10) и нажмите кнопку «Установить размер».
- После этого динамически появятся поля для ввода коэффициентов матрицы и свободных членов.

3. Решение системы:

- Заполните поля для ввода значениями коэффициентов и свободных членов.
- Нажмите кнопку «Решить» для вычисления решений системы (рис. 2).

– В случае успешного решения результаты будут отображены в диалоговом окне. Если система не имеет уникального решения, программа выдаст соответствующее сообщение об ошибке (рис. 3).

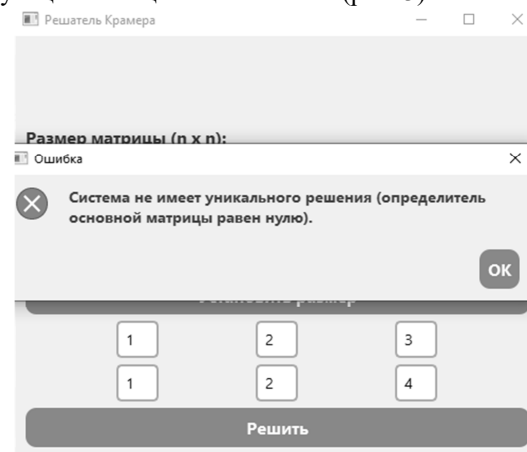


Рис. 3. Диалоговое окно – соответствующее сообщение об ошибке

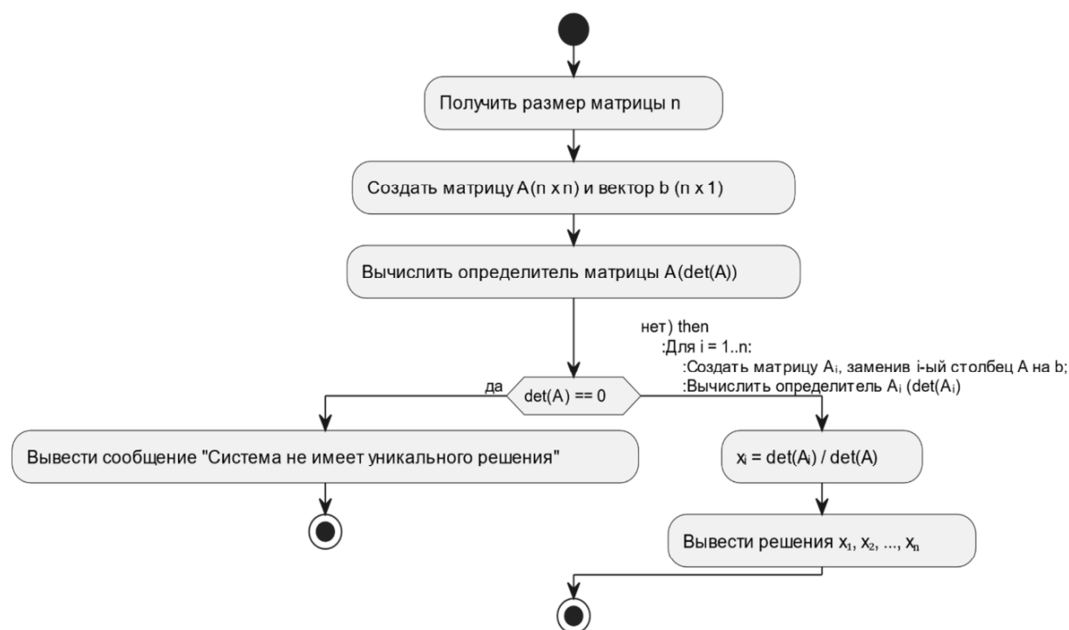


Рис. 4. Блок-схема кода

Заключение

Разработанная программа представляет собой удобный инструмент для решения систем линейных уравнений методом Крамера. Она предоставляет пользователю интуитивно понятный интерфейс, созданный с помощью PyQt6, что упрощает процесс ввода данных и получения результатов. Программа может быть полезна как для студентов, изучающих линейную алгебру, так и для специалистов, работающих с математическими задачами. Обработка ошибок и исключений в программе способствует повышению ее надежности и удобства в использовании.

Перспективы развития

В дальнейшем программу можно расширить следующими функциями:

1. Поддержка более сложных систем уравнений: Включение методов решения систем с комплексными коэффициентами или неравенствами.
2. Графическое отображение результатов: визуализация решений в виде графиков или диаграмм, что поможет пользователям лучше понимать и анализировать результаты.
3. Расширение функционала: добавление возможности решать системы уравнений другими методами (например, методом Гаусса), что повысит гибкость программы.
4. Улучшение пользовательского интерфейса: введение дополнительных инструкций по вводу данных и визуальных подсказок для облегчения работы с программой.

Литература

1. Рахимов, А. А. Компьютерная система Maple как средство формирования творческой самостоятельности в обучении высшей математике студентов технических вузов в условиях кредитной технологии обучения / А. А. Рахимов // Вестник Таджикского национального университета. Серия естественных наук. – 2017. – № 1–4. – С. 57–60.
2. Рахимов, А. А. Компьютерное моделирование как один из способов повышения эффективности обучения по высшей математике в техническом вузе / А. А. Рахимов // Вестник Костромского государственного университета. Серия: Педагогика. Психология. Социокинетика. – 2023. – Т. 29, № 2. – С. 132–143.
3. Рахимов, А. А. Компьютерное моделирование как один из способов математической подготовки студентов в техническом вузе / А. А. Рахимов, Ф. Джалилов, М. М. Комилов // Психолого-педагогические и лингвокультурологические исследования: от теории к практике : материалы XII Всероссийской научно-практической конференции (Москва, 08 июня 2023 г.). – Москва : ООО «Академ-пресс», 2023. – С. 122–128. – EDN AWQSGO.
4. Рахимов, А. А. Методика моделирования процесса нахождения приближенных значений определённого интеграла с помощью формулы прямоугольников с применением программы Javascript / А. А. Рахимов, А. А. Умаров, Х. К. Мухаббатов // Вестник

Педагогического университета. Серия 2: Педагогики и психологии, методики преподавания гуманитарных и естественных дисциплин. – 2023. – № 3 (17). – С. 155–161.

5. Гуломнабиев, С. Г. Оид ба истифодаи барномаи Maple дар раванди таълим / С. Г. Гуломнабиев // Вестник Педагогического университета. Серия 2: Педагогики и психологии, методики преподавания гуманитарных и естественных дисциплин. – 2023. – № 4. – 1(18). – С. 148–157.

6. Нугмонов, М. Методическое обеспечение самостоятельной работы студентов по высшей математике в техническом вузе в условиях кредитной технологии обучения / М. Нугмонов, А. А. Рахимов // Вестник Педагогического университета. – 2013. – № 5. – 2(54). – С. 200–205.

8. Аминова, З. А. Методика использования занимательных заданий в процессе обучения математике в 5 классе / З. А. Аминова, А. А. Рахимов // Вестник Челябинского государственного педагогического университета. – 2012. – № 7. – С. 11–20.

9. Абдуллоев, Н. С. Дифференциация обучения высшей математике при интеграции в него информационных технологий в технических вузах / Н. С. Абдуллоев, А. А. Рахимов // Вестник Череповецкого государственного университета. – 2012. – № 3-2(41). – С. 137–139.

10. Численные методы. Примеры и задачи. Учебно-методическое пособие по курсам «Информатика» и «Вычислительная математика» / составитель: Ф. Г. Ахмадиев, Ф. Г. Габбасов, Л. Б. Ермолаева, И. В. Маланичев. – Казань : КГАСУ, 2017. – 107 с.

11. Глазырина, Л. Л. Введение в численные методы : учебное пособие / Л. Л. Глазырина, М. М. Карчевский. – Казань : Казанский университет, 2017. – 122 с.

12. Киясов, С. Н. Дифференциальные уравнения. Основы теории, методы решения задач : учебное пособие / С. Н. Киясов, В. В. Шурыгин. – Казань : Казанский федеральный университет, 2011. – 112 с.

13. Рахимов, А. А. Использование компьютерного моделирования в процессе обучения математике и физике для подготовке студентов в техническом вузе / А. А. Рахимов // Мировые стратегии развития науки и образования в новой реальности: междисциплинарные исследования : материалы I Международной научно-практической конференции (Москва, 29 января 2024 г.). – Москва : Издательство «Научная парадигма», 2024. – С. 157–162.

14. Рахимов, А. А. Использование компьютерного моделирования AUTOCAD в образовательном процессе для студентов технических направлений вуза / А. А. Рахимов // Наука и практика в образовании: электронный научный журнал. – 2024. – Т. 5, № 2. – С. 43–48. – DOI 10.54158/27132838_2024_5_2_43. – EDN CMTUEX.

15. Рахимов, А. А. Компьютерное моделирование один из способов повышения эффективности обучения по высшей математике в техническом вузе / А. А. Рахимов // Вестник Бохтарского государствен-

ного университета имени Носира Хусрава. Серия гуманитарных и экономических наук. – 2023. – № 1–1(107). – С. 147–159. – EDN EISATE.

16. Селиванова, Э. Т. Методика обучения основам компьютерного моделирования в педагогическом

вузе и школе : диссертация на соискание ученой степени кандидата педагогических наук [Текст] / Э. Т. Селиванова. – Новосибирск, 2000. – 144 с.

A.A. Rakhimov, S.K. Ismoilova

Polytechnic Institute of Tajik Technical University named after the academician M. Osimi

USING COMPUTER SIMULATION FOR SOLVING SYSTEMS OF ALGEBRAIC EQUATIONS BY CRAMER'S RULE

This article considers a report on the development of a program for solving systems of linear equations by Cramer's rule.

As part of this work, a program was developed for solving systems of linear equations by Cramer's rule using the Python programming language. This method is used in linear algebra to find solutions to square systems of equations, provided that the determinant of the coefficient matrix is not equal to zero. The program is implemented in Python using the PyQt6 library, which provides tools for creating a graphical user interface. This allows users to easily enter data and get results without having to interact with the code.

Cramer's rule, Python, computer modeling, system of linear equations, graphical interface.