УДК КОД

ЭФФЕКТИВНОЕ ПОСТРОЕНИЕ МАТРИЦ АДАМАРА НА ОСНОВЕ МЕТОДА ПЕЙЛИ: МЕТОДИЧЕСКАЯ СХЕМА И УЧЕБНОЕ ПРИЛОЖЕНИЕ

Евдокимов Иван Дмитриевич, студент, направление подготовки 02.03.02 Фундаментальная информатика и информационные технологии, Оренбургский государственный университет, Оренбург
e-mail: vancouver.evdokimov@yandex.ru

Носов Виталий Валерьевич, кандидат физико-математических наук, доцент, доцент кафедры математики и цифровых технологий, Оренбургский государственный университет, Оренбург
e-mail: puncker1978@mail.ru

***Аннотация**: В работе рассматривается эффективное построение матриц Адамара на основе метода Пейли, используемого в теории кодирования. Предложена методическая схема, в которой вычисления опираются на свойства символа Лежандра и построение матрицы Джекобстола. Показано, что строки матрицы Джекобстола образуют циклические сдвиги, что позволяет существенно упростить построение: вся матрица определяется одной строкой. Данное свойство переносится и на матрицу Адамара, получаемую вычитанием из матрицы Джекобстола единичной матрицы, благодаря чему значительная часть построения сводится к наглядным циклическим преобразованиям. Каждый этап сопровождается промежуточными выводами и пояснениями, а разработанное учебное приложение визуализирует процесс пошагово. Представленный подход делает метод Пейли более доступным для студентов и может быть использован в образовательных целях при изучении дисциплины «Теория кодирования».*

***Ключевые слова**: теория кодирования, матрица Адамара, метод Пейли, символ Лежандра, матрица Джекобстола, квадратичные вычеты, циклические сдвиги, обучающее приложение, методическая схема, построение кодов.*

***Для цитирования**: Евдокимов И. Д., Носов В. В. Эффективное построение матриц Адамара на основе метода Пейли: методическая схема и учебное приложение // Оренбург – 2025 г.*

EFFICIENT CONSTRUCTION OF HADAMARD MATRICES BASED ON THE PALEY METHOD: METHODOLOGICAL SCHEME AND EDUCATIONAL APPLICATION

Evdokimov Ivan Dmitrievich, Undergraduate Student, Program 02.03.02 *Fundamental Informatics and Information Technologies*, Orenburg State University, Orenburg
e-mail: vancouver.evdokimov@yandex.ru

Nosov Vitaliy Valerievich, Candidate in Physics and Mathematics, Associate Professor, Department of Mathematics and Digital Technologies, Orenburg State University, Orenburg
e-mail: puncker1978@mail.ru

***Abstract**: This paper discusses the efficient construction of Hadamard matrices based on the Paley method, which is widely used in coding theory. A methodological scheme is proposed in which the calculations rely on the properties of the Legendre symbol and the construction of the Jacobsthal matrix. It is shown that the rows of the Jacobsthal matrix form cyclic shifts, which significantly simplifies the construction: the entire matrix is determined by a single row. This property also extends to the Hadamard matrix, obtained by subtracting the identity matrix from the Jacobsthal matrix, whereby a substantial part of the construction is reduced to illustrative cyclic transformations. Each stage is accompanied by intermediate results and explanatory notes, while the developed educational application visualizes the process step by step. The proposed approach makes the Paley method more accessible to students and can be applied for educational purposes in the study of coding theory.*

***Keywords**: coding theory, Hadamard matrices, Paley method, Legendre symbol, Jacobsthal matrix, quadratic residues, cyclic shifts, educational application, methodological scheme, code construction.*

***For citation:** Evdokimov I. D., Nosov V. V. Efficient Construction of Hadamard Matrices Based on the Paley Method: Methodological Scheme and Educational Application. Orenburg, 2025.*

Введение

В современной теории кодирования значительное внимание уделяется методам построения ортогональных матриц, обладающих хорошими свойствами с точки зрения помехоустойчивости и избыточности. Одним из наиболее известных объектов этого направления являются матрицы Адамара, элементы которых принимают значения $+1$ и -1 и строки которых взаимно ортогональны. Эти матрицы находят применение не только в кодировании, но и в задачах обработки сигналов, криптографии и комбинаторике. В частности, на их основе строятся коды Адамара, обладающие высокой корректирующей способностью.

Существует несколько способов построения матриц Адамара различных порядков. Среди них особое место занимает метод Пейли, позволяющий конструировать матрицы на основе числовых свойств простых чисел и символа Лежандра. Данный метод является не только конструктивным, но и обладает важной педагогической ценностью: он связывает объекты алгебры и теории чисел с задачами кодирования. В классическом изложении метод Пейли опирается на построение вспомогательной матрицы Джекобстола, элементы которой определяются через значения символа Лежандра. Впоследствии из этой матрицы формируется матрица Адамара требуемого порядка.

Однако традиционные описания метода Пейли зачастую сводятся к приведению готовых формул или к использованию вычислительных средств, что затрудняет понимание внутренней структуры построения. При этом сам процесс обладает рядом закономерностей, которые позволяют существенно упростить объяснение. Так, в работе обращается внимание на важное свойство матрицы Джекобстола: её строки образуют циклические сдвиги, что напрямую вытекает из способа задания элементов через разность индексов. Это наблюдение не только обосновывает упрощённое построение всей матрицы на основе одной строки, но и переносится на матрицу Адамара, получаемую в результате вычитания единичной матрицы.

Цель данной работы – предложить академическую, пошаговую схему построения матриц Адамара на основе метода Пейли, в которой каждое преобразование сопровождается промежуточными выводами и пояснениями. Такой подход позволяет сделать материал более наглядным и доступным для студентов, изучающих дисциплину «Теория кодирования». Дополнительно разработано учебное приложение, реализующее предложенную схему и демонстрирующее процесс построения в интерактивной форме. Представленные результаты ориентированы на методическую поддержку учебного процесса и могут быть использованы при проведении лекций и практических занятий по теории кодирования.

Теоретические основы метода Пейли и его применение

Метод Пейли I (при $p \equiv 3(mod\,4)$) опирается на использование символа Лежандра. Он определяется для нечётного простого числа p и целого числа a следующим образом:

$$\left(\frac{a}{p}\right) = \begin{cases} 0, & p|a \\ 1, & \exists\, x \in \mathbb{Z} :\ x^2 \equiv a (\mathrm{mod}\, p) \\ -1, & \nexists\, x \in \mathbb{Z} :\ x^2 \equiv a (\mathrm{mod}\, p) \end{cases}$$

Значение $\left(\frac{a}{p}\right)$ показывает, является ли a квадратичным вычетом по модулю p. При этом, для построений применяется ряд фундаментальных свойств, упрощающих вычисление символа Лежандра на практике:

1) Критерий Эйлера (свойство симметрии)

$$\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} (\mathrm{mod}\, p)$$

2) Свойство для квадратов

$$\left(\frac{a^2}{p}\right) = 1, \qquad p \nmid a$$

3) Значение для $a = -1$

$$\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}} = \begin{cases} 1, & p \equiv 1 (\mathrm{mod}\, 4) \\ -1, & p \equiv 3 (\mathrm{mod}\, 4) \end{cases}$$

4) Значение для $a = 2$

$$\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}} = \begin{cases} 1, & p \equiv \pm 1 (\mathrm{mod}\, 8) \\ -1, & p \equiv \pm 3 (\mathrm{mod}\, 8) \end{cases}$$

5) Свойство редукции

$$\left(\frac{a}{p}\right) = \left(\frac{a \bmod p}{p}\right)$$

6) Мультипликативность

$$\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$$

7) Закон квадратичной взаимности

$$\left(\frac{p}{q}\right) = (-1)^{\frac{(p-1)}{2}\cdot\frac{(q-1)}{2}}\left(\frac{q}{p}\right)$$

При академических подсчётах без использования ЭВМ данные свойства значительно упрощают вычисления, сводя задачу к известным случаям.

В методе Пейли I символ Лежандра используется для построения матрицы Джекобстола порядка p, элементы которой являются символами Лежандра от разности индексов:

$$J_p(i,j) = \left(\frac{i-j}{p}\right), 0 \le i,j < p$$

Далее с использованием матрицы Джекобстола строится H матрица Адамара порядка $p+1$:

$$H = \begin{pmatrix} 1 & \mathbf{1}^T \\ \mathbf{1} & J_p - I_p \end{pmatrix},$$

где $\mathbf{1}$ – вектор из p единиц, а Ip – единичная матрица, соответствующая порядком матрице Джекобстола.

Матрица H, построенная по описанному алгоритму обладает фундаментальным свойством:

$$HH^T = (p+1)I_{p+1}$$

Это равенство означает ортогональность строк и столбцов H. Таким образом, матрица H является матрицей Адамара.

Практическое приложение матриц Адамара в рамках теории кодирования выражается, в частности, в построении кодов Адамара через матрицу, в которой осуществляются замены $1 \to 0$ и $-1 \to 1$.

Ключевым образом, возможно построение 3 вариантов кода Адамара.

Для получения первого усечённого кода Адамара с параметрами количества слов, длины и расстояния $(n-1,\ n,\ \frac{n}{2})$ необходимо отбросить первый столбец матрицы Адамара, состоящий целиком из нулей. Словами, как и в остальных вариациях, будут являться полученные строки.

Второй усечённый вариант с дополнениями возможного кода Адамара обладает тройкой параметров $(n-1,\ 2n,\ \frac{n}{2}-1)$ и получается добавлением побитовой инверсии каждой строки исходной матрицы с отброшенным нуль-столбцом. Таким образом, код выполняет функцию расширения множества слов по сравнению с первой вариацией.

Если в начале не отбрасывать нуль столбец, а побитово проинвертировать исходные строки и добавить их, то мы получим третий возможный код Адамара (полный) с параметрами $(n,\ 2n,\ \frac{n}{2})$.

Циклические свойства матриц Джекобстола и Адамара

Важным свойством матрицы Джекобстола является её циклическая структура. Для формализации введём оператор циклического сдвига вправо. Пусть дана строка вида $x = (x_0, x_1, \dots, x_{n-2}, x_{n-1})$. Циклическим сдвигом вправо на один элемент будем называть преобразование

$$T(x) = (x_{n-1}, x_0, x_1, \dots, x_{n-2}),$$

при котором последний элемент строки перемещается в начало, а остальные элементы сдвигаются на одну позицию вправо. Очевидно, многократное применение этого оператора соответствует сдвигу на большее число позиций: $T^k(x)$ даёт сдвиг вправо на k элементов.

Рассмотрим теперь матрицу Джекобстола порядка q, где $q \equiv 3 (mod\ 4)$. Её элементы задаются формулой

$$J_{ij} = \begin{cases} 0, & i = j \\ \lambda(j-i), & i \neq j \end{cases} \quad i, j \in \mathbb{Z}_q$$

где $\chi(\cdot)$ обозначает квадратичный характер (символ Лежандра). Для фиксированного i строка матрицы J состоит из значений $\chi(j-i)$ при $j = 0,1, \dots, q-1$. Если рассмотреть строку с индексом $i+1$, то её элементы будут иметь вид $J_{i+1,j} = \chi\big(j-(i+1)\big)$. Переписывая это выражение, получаем равенство $J_{i+1,j} = \chi\big((j-1)-i\big) = J_{i,j-1}$. Иными словами, каждая следующая

строка получается из предыдущей действием оператора T, то есть представляет собой циклический сдвиг вправо на один элемент.

Отсюда следует, что вся матрица Джекобстола полностью определяется своей первой строкой. Достаточно вычислить значения $\chi(k)$ для $k = 0,1, \dots, q-1$, после чего все остальные строки могут быть получены последовательными сдвигами T. Тогда с учётом определения циклического сдвига первой строки можно записать матрицу Джекобстола в следующем виде:

$$J = \begin{pmatrix} x \\ T(x) \\ \dots \\ T^{q-1}(x) \end{pmatrix}$$

Таким образом, построение матрицы существенно упрощается: вместо $q(q-1)$ вычислений квадратичного характера требуется лишь один его проход по всем элементам поля, а оставшаяся часть восстанавливается комбинаторно.

Те же самые свойства переносятся на матрицу Адамара, строящуюся по методу Пейли. Возвращаясь к определению, определяем её через следующую блочную структуру:

$$H = \begin{pmatrix} 1 & \mathbf{1}^T \\ \mathbf{1} & J - I_q \end{pmatrix},$$

где $\mathbf{1}$ обозначает вектор из единиц длины q. В нижнем правом блоке $J - I_q$ циклические свойства сохраняются полностью: строки этого блока так же порождаются одна из другой сдвигом вправо на один элемент. Изменению подвергаются лишь диагональные позиции, которые в силу вычитания единичной матрицы принимаются значение -1. Следовательно, для получения всей матрицы Адамара (не считая первой строки и первого столбца, которые состоят из единиц) достаточно вычислить первую строку блока J, затем скорректировать её диагональный элемент и далее восстановить остальные строки с помощью оператора T.

Таки образом, использование циклического сдвига позволяет существенно сократить объем вычисление и одновременно наглядно продемонстрировать внутреннюю структуру как матрицы Джекобстола, так и матрицы Адамара. Это упрощение полезно не только при практическом построении таких матриц, но и при их теоретическом анализе, поскольку оно сразу выявляет глубинное свойство ортогональности, связанное с равномерным распределения квадратичных вычетов.

Определение алгоритмической схемы построения матрицы Адамара через формализацию работ с символом Лежандра и циклическими свойствами матриц Джекобстола и Адамара.

Построение матрицы Адамара по методу Пейли сводится к определению матрицы Джекобстола и последующему использованию её циклической структуры. Как отмечалось выше, вся матрица Джекобстола восстанавливается из одной единственной строки с помощью оператора циклического сдвига вправо. Следовательно, ключевым вопросом становится вычисление значений квадратичного характера $\chi(k)$ для всех $\mathrm{k} = 0,1,\dots,\mathrm{q}-1$, то есть символа Лежандра $\left(\frac{k}{q}\right)$. Таким образом, алгоритм построения матрицы Адамара фактически распадается на два взаимосвязанных подалгоритма: вычисление первой строки матрицы Джекобстола через символы Лежандра и воспроизведение остальных строк посредством циклических сдвигов.

Для того чтобы алгоритм имел строгую и воспроизводимую структуру, необходимо задать однозначную схему вычисления символа Лежандра. Она основывается на совокупности классических свойств: критерии Эйлера, значении символа для -1 и для 2, свойстве мультипликативности, свойстве редукции и законе квадратичной взаимности. В практическом применении эти свойства организуются в последовательность, которая гарантирует снижение сложности задачи на каждом шаге.

Начинается вычисление всегда с редукции: аргумент aзаменяется на его остаток по модулю p. В случае, если остаток равен нулю, символ немедленно обращается в ноль. Если остаток отрицателен, его знак убирается с помощью специальной формулы для $\left(\frac{-1}{p}\right)$, и этот вклад фиксируется как дополнительный множитель $(-1)^{\frac{p-1}{2}}$. Далее из аргумента выносятся все степени числа 2, и каждая такая редукция учитывается множителем $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$. После этих операций числитель становится нечётным и положительным.

На этом этапе возможны два типа раннего завершения. Если по самому виду аргумента очевидно, что он является квадратом по модулю p, например, когда он совпадает с целым квадратом или принадлежит заранее вычисленному множеству квадратичных вычетов для малых модулей, то значение символа сразу фиксируется равным единице. Другой вариант заключается в применении критерия Эйлера: при достаточно малых числах или при желании ускорить вычисления вместо дальнейших преобразований можно воспользоваться равенством $\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} (mod\, p)$, что позволяет завершить процесс непосредственно через модульное возведение в степень.

Если аргумент после редукции и удаления факторов 2 не является ни единицей, ни нулём, и не даёт немедленной развязки, применяется закон квадратичной взаимности. Он устанавливает равенство

$$\left(\frac{a}{p}\right) = (-1)^{\frac{a-1}{2}\cdot\frac{p-1}{2}}\left(\frac{p}{a}\right),$$

которое, во-первых, вносит дополнительный множитель в текущий знак, а во-вторых, меняет местами роли числителя и модуля, после чего производится очередная редукция по новому модулю. Важно, что в результате этого шага модуль строго уменьшается, и последовательность преобразований не может продолжаться бесконечно: в конечном итоге всегда достигается случай $a = 1$ или $a \equiv 0(mod\, p)$.

Таким образом, вычисление символа Лежандра превращается в процесс, где на каждом шаге применяется либо упрощающее правило (для -1, для 2 или для квадрата), либо процедура взаимности с последующей редукцией. При необходимости в любой момент можно воспользоваться критерием Эйлера, если размеры чисел позволяют выполнить возведение в степень быстрее, чем продолжить взаимные преобразования.

Соединив этот подалгоритм с циклической структурой матрицы Джекобстола, получаем полный алгоритм построения матрицы Адамара. Сначала значения символа Лежандра вычисляются по изложенной схеме для всех чисел от 0 до $q - 1$, формируя первую строку матрицы Джекобстола. Затем вся матрица восстанавливается из этой строки с помощью итерации оператора циклического сдвига вправо, а на последнем этапе в матрицу вводятся первая строка и первый столбец из единиц и корректируются диагональные элементы блока $J - I_q$. Благодаря такой организации удаётся минимизировать количество прямых обращений к символу Лежандра и свести задачу к одной вычислительной процедуре плюс простым комбинаторным перестановкам.

Для наглядности рассмотрим случай $q = 7$. Так как $7 \equiv 3(mod\, 4)$, по методу Пейли гарантируется существование матрицы Адамара порядка $n = q + 1 = 8$.

Начнём с вычисления первой строки матрицы Джекобстола. Для этого требуется определить значения $\left(\frac{k}{7}\right)$ для $k = 0,1,2,3,4,5,6$. Символ при нуле равен нулю. Далее применяем подалгоритм.

- Для $k = 1$ результат равен единице, так как 1 – очевидный квадрат.
- Для $k = 2$ используем известную формулу $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$. При $p = 7$ получаем $\frac{7^2-1}{8} = 6$, что чётно, следовательно $\left(\frac{2}{7}\right) = 1$.
- Для $k = 3$ удобно применить закон квадратичной взаимности:

$\left(\frac{3}{7}\right) = (-1)^{\frac{3-1}{2}\cdot\frac{7-1}{2}}\left(\frac{7}{3}\right)$. Так как показатель равен $\frac{2}{2}\cdot\frac{6}{2}$, он нечётен, множитель равен -1. Далее $\left(\frac{7}{3}\right) = \left(\frac{1}{3}\right)$, а символ $\left(\frac{1}{3}\right) = 1$. Значит, $\left(\frac{3}{7}\right) = -1$.

- Для $k = 4$ достаточно заметить, что $4 = 2^2$, следовательно $\left(\frac{4}{7}\right) = 1$.
- Для $k = 5$ снова применим взаимность:
  $\left(\frac{5}{7}\right) = (-1)^{\frac{5-1}{2}\cdot\frac{7-1}{2}}\left(\frac{7}{5}\right)$. Здесь показатель равен $\frac{4}{2}\cdot\frac{6}{2} = 6$, множитель равен $+1$. Остаётся $\left(\frac{7}{5}\right) = \left(\frac{2}{5}\right)$. По формуле для числа 2 при $p = 5$ получаем $-1^{\frac{25-1}{8}} = (-1)^3 = -1$. Следовательно, $\left(\frac{5}{7}\right) = -1$.
- Для $k = 6$ заметим, что $6 \equiv -1(mod\ 7)$. Поэтому $\left(\frac{6}{7}\right) = \left(\frac{-1}{7}\right) = (-1)^{\frac{7-1}{2}}(-1)^3 = -1$.

  Таким образом, первая строка матрицы Джекобстола имеет вид

$$r^{(0)} = (0,1,1,-1,1,-1,-1).$$

Дальнейшее построение матрицы выполняется исключительно с помощью оператора циклического сдвига вправо: каждая следующая строка есть результат применения этого оператора к предыдущей. В частности, вторая строка будет равна $r^{(1)} = (-1,0,1,1,-1,1,-1)$, третья – $r^{(2)} = (-1,-1,0,1,1,-1,1)$, и так далее до $r^{(6)}$. В результате формируется вся матрица Джекобстола.

$$J = \begin{pmatrix} 0 & 1 & 1 & -1 & 1 & -1 & -1 \\ -1 & 0 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & 0 & 1 & 1 & -1 & 1 \\ 1 & -1 & -1 & 0 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 0 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & 0 \end{pmatrix}$$

Чтобы перейти к матрице Адамара порядка 8, к этой структуре добавляется первая строка и первый столбец из единиц, а на диагонали блока J нули заменяются на -1. Получается матрица, у которой символами $*$ заполнены циклически порождённые строки матрицы Джекобстола с учётом коррекции диагонали вычитанием единичной матрицы.

$$H = \begin{pmatrix} 1 & 1 & \dots & 1 \\ 1 & * & \dots & * \\ \dots & \dots & \dots & \dots \\ 1 & * & \dots & * \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & -1 & 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 & -1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & 1 & -1 & -1 & -1 \end{pmatrix}$$

Учебное приложение для визуализации процесса

В рамках работы была создана программная реализация [https://github.com/EvdokimovID/Coding-theory-scientific-article], позволяющая наглядно проследить этапы вычислений и построения структур, связанных с матрицами и кодами Адамара. Разработанный интерактивный инструмент представляет собой учебное приложение в формате ноутбука, ориентированного на визуализацию и экспериментальное изучение рассматриваемой схемы.

Основная идея состоит в том, чтобы объединить несколько последовательных шагов анализа – от построения исходных матриц до генерации кодовых последовательностей – и предоставить пользователю удобные средства для их наблюдения и сравнения. Такой подход позволяет уйти от исключительно теоретического изложения и сделать акцент на процессуальной стороне работы, показывая, как абстрактные конструкции проявляют себя на практике в вычислительной среде.

Пайплайн приложения включает несколько ключевых модулей. В первых ячейках сосредоточен код, реализующий базовые функции построения:

- *trace_legendre* отвечает за вычисление значений символа Лежандра и построение соответствующей матрицы, служащей фундаментом для последующих шагов;
- *jacobsthal_matrix* формирует матрицу Джекобстола, позволяющую рассматривать переход к более сложным структурам и обеспечивающую связку между исходными элементами и будущими построениями;
- *hadamard_matrix* реализует рекурсивный алгоритм построения матрицы Адамара, в основе которого лежит принцип ортогональности строк и симметричного расширения.

Эти шаги выделены в отдельные блоки, что обеспечивает структурированность и возможность независимого запуска каждого из них. Такой подход удобен для студентов: они могут изолированно исследовать отдельные этапы, а затем проследить, как они соединяются в общую систему.

На следующем уровне реализован модуль интеграции, где отдельные процедуры объединяются в общий вычислительный процесс. Именно здесь пользователь получает возможность запускать цепочку преобразований целиком, наблюдая за последовательным формированием структур и их свойств. Этот модуль играет ключевую роль: он позволяет воспринимать материал не как набор разрозненных операций, а как единый алгоритм.

Особое внимание уделено блоку визуализации. Матрицы выводятся в наглядном виде с возможностью динамического изменения параметров. Для этого используется интерактивный элемент управления – слайдер, который позволяет изменять размерность задачи и мгновенно наблюдать за изменением результатов. Такая визуализация значительно облегчает восприятие, поскольку

позволяет непосредственно увидеть взаимосвязь между параметрами и получаемыми матрицами.

Завершающий компонент пайплайна – единый интерфейс запуска. В последней ячейке сосредоточена интерактивная кнопка, инициирующая работу всех модулей приложения. Благодаря этому пользователь может, не вдаваясь в технические детали, сразу получить полный набор результатов, необходимых для анализа. Это особенно важно в учебных целях, так как позволяет сконцентрироваться на понимании сути явлений, а не на технической стороне вычислений.

Таким образом, созданное приложение выполняет сразу несколько функций. Оно служит иллюстративным материалом, наглядно демонстрирующим работу ключевых алгоритмов; выступает в роли тренажёра, где студенты могут самостоятельно изменять параметры и исследовать поведение системы; наконец, оно является вспомогательным инструментом для преподавателя, упрощающим объяснение материала за счёт интерактивных визуализаций.

Фрагмент итогового вывода отображён на рисунках 1-3.

```
q = 7 (q ≡ 3 mod 4). Первая строка r^(0):
[0, 1, 1, -1, 1, -1, -1]

--- k = 0, (k/q) = 0 ---
  Определение: a=0, p=7, знак=1; (0/q)=0

--- k = 1, (k/q) = 1 ---
  Редукция: a ← a mod p: a=1, p=7, знак=1; Исходное a=1
  Тривиальный случай: (1/p)=1: a=1, p=7, знак=1;

--- k = 2, (k/q) = 1 ---
  Редукция: a ← a mod p: a=2, p=7, знак=1; Исходное a=2
  Фактор 2: (2/p) = +1: a=1, p=7, знак=1; (p^2-1)/8 чётно
  Тривиальный случай: (1/p)=1: a=1, p=7, знак=1;

--- k = 3, (k/q) = -1 ---
  Редукция: a ← a mod p: a=3, p=7, знак=1; Исходное a=3
  Взаимность: множитель -1: a=3, p=7, знак=-1; Показатель нечётен
  Смена ролей и редукция: (p mod a, a): a=1, p=3, знак=-1;
  Тривиальный случай: (1/p)=1: a=1, p=3, знак=-1;

--- k = 4, (k/q) = 1 ---
  Редукция: a ← a mod p: a=4, p=7, знак=1; Исходное a=4
  Фактор 2: (2/p) = +1: a=2, p=7, знак=1; (p^2-1)/8 чётно
  Фактор 2: (2/p) = +1: a=1, p=7, знак=1; (p^2-1)/8 чётно
  Тривиальный случай: (1/p)=1: a=1, p=7, знак=1;

--- k = 5, (k/q) = -1 ---
  Редукция: a ← a mod p: a=5, p=7, знак=1; Исходное a=5
  Взаимность: множитель +1: a=5, p=7, знак=1; Показатель чётен
  Смена ролей и редукция: (p mod a, a): a=2, p=5, знак=1;
  Фактор 2: (2/p) = -1: a=1, p=5, знак=-1; (p^2-1)/8 нечётно
  Тривиальный случай: (1/p)=1: a=1, p=5, знак=-1;

--- k = 6, (k/q) = -1 ---
  Редукция: a ← a mod p: a=6, p=7, знак=1; Исходное a=6
  Фактор 2: (2/p) = +1: a=3, p=7, знак=1; (p^2-1)/8 чётно
  Взаимность: множитель -1: a=3, p=7, знак=-1; Показатель нечётен
  Смена ролей и редукция: (p mod a, a): a=1, p=3, знак=-1;
  Тривиальный случай: (1/p)=1: a=1, p=3, знак=-1;
```

Рисунок 1 – Трассирование поэтапных вычислений символа Лежандра

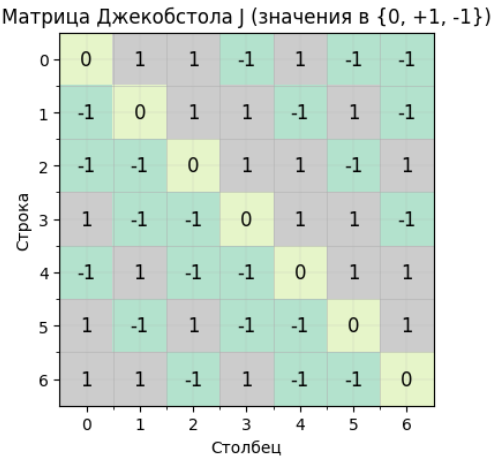

Рисунок 2 – матрица Джекобстола

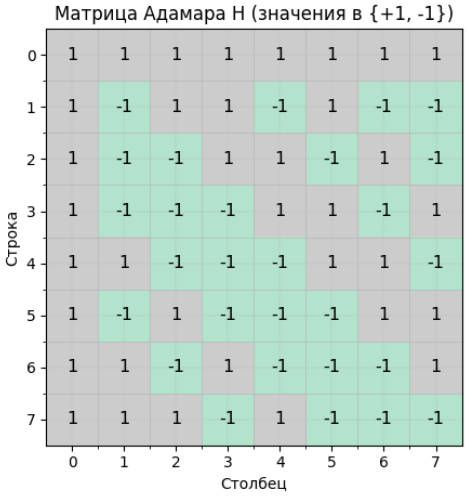

Рисунок 3 – матрица Адамара

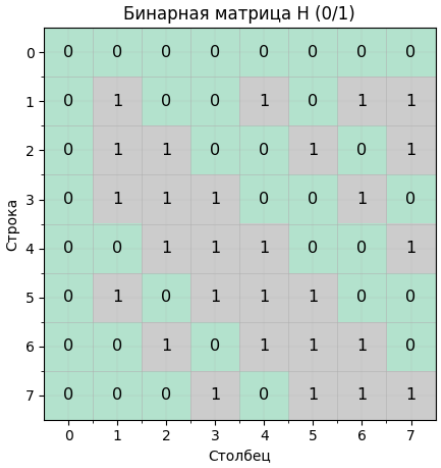

Рисунок 4 – бинарная матрица Адамара

C1_(n-1,n,n/2): форма (8, 7), d_min = 4

	index	bits	weight
0	0	0000000	0
1	1	1001011	4
2	2	1100101	4
3	3	1110010	4
4	4	0111001	4
5	5	1011100	4
6	6	0101110	4
7	7	0010111	4

Рисунок 5 – фрагмент кодов Адамара

Заключение

Предложенная методическая схема демонстрирует, что построение матриц Адамара на основе метода Пейли может быть представлено в наглядной и воспроизводимой форме. Использование свойств символа Лежандра и циклической структуры матрицы Джекобстола позволило существенно упростить процесс вычислений и одновременно подчеркнуть внутреннюю закономерность построений.

Разработанное учебное приложение выполняет роль практического инструмента, позволяющего визуализировать отдельные этапы алгоритма и проследить переход от числовых характеристик к конечным кодовым структурам. Такой подход способствует лучшему усвоению материала и может быть применён как в учебных курсах по теории кодирования, так и в самостоятельной работе студентов.