

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |

**Институт** Информационных Технологий

**Кафедра** Вычислительной Техники

**ПРАКТИЧЕСКАЯ РАБОТА №3**

**по дисциплине**

**«Теория принятия решений»**

**Метод анализа иерархий**

Студент группы: ИКБО-41-23 \_\_Трофимов Андрей\_\_ *(Ф. И.О. студента)*

Преподаватель \_\_Железняк Л.М.\_\_

*(Ф.И.О. преподавателя)*

|  |  |  |
| --- | --- | --- |
|  |  |  |

Москва 2025

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc130978911)

[1 МЕТОД АНАЛИЗА ИЕРАРХИЙ 4](#_Toc130978912)

[1.1 Постановка задачи 4](#_Toc130978913)

[1.2 Представление проблемы в виде иерархии 4](#_Toc130978914)

[1.3 Установка приоритетов критериев 5](#_Toc130978915)

[1.4 Синтез приоритетов 5](#_Toc130978916)

[1.5 Согласованность локальных приоритетов 14](#_Toc130978917)

[3.6 Синтез альтернатив 20](#_Toc130978918)

[3.7 Вывод 21](#_Toc130978919)

[1.8 Результаты работы программы 22](#_Toc130978920)

[ЗАКЛЮЧЕНИЕ 23](#_Toc130978921)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ 24](#_Toc130978922)

[ПРИЛОЖЕНИЯ 25](#_Toc130978923)

ВВЕДЕНИЕ

Метод анализа иерархий (МАИ) является мощным инструментом, применяемым для решения задач, связанных с многокритериальной оценкой и принятием решений. Он используется для упорядочивания множества альтернатив по определённым критериям, что позволяет принять обоснованное решение в условиях неопределённости и сложности. МАИ основывается на сравнении элементов пары, что помогает учитывать все возможные предпочтения и важность факторов для каждого варианта.

Этот метод широко используется в различных областях, таких как управление проектами, бизнес-анализ, выбор оптимальных стратегий, а также в разработке систем поддержки принятия решений. Например, в задачах выбора наилучшей модели автомобиля, оптимизации производственных процессов, распределении ресурсов или планировании.

Применение МАИ позволяет структурировать и формализовать процесс принятия решений, минимизируя субъективность и учитывая различные аспекты, что делает его полезным инструментом в задачах с множеством критериев, таких как выбор поставщика, оценка финансовых показателей или даже в личных предпочтениях.

1 МЕТОД АНАЛИЗА ИЕРАРХИЙ

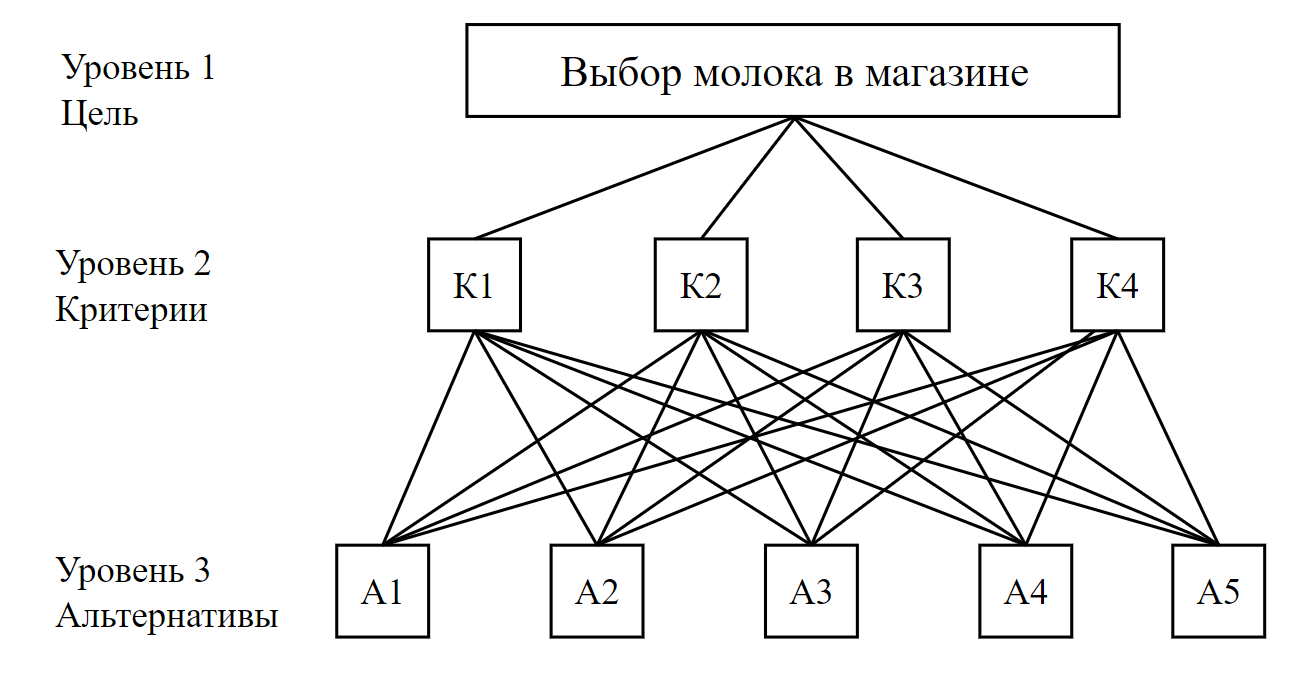
## **1.1 Постановка задачи**

Задача практической работы: выбор молока в магазине.

## **1.2 Представление проблемы в виде иерархии**

Первый этап – представление проблемы в виде иерархии или сети. В простейшем случае, иерархия строится, начиная с цели, которая помещается в вершину иерархии. Через промежуточные уровни, на которых располагаются критерии и от которых зависят последующие уровни, к самому низкому уровню, который содержит перечень альтернатив.

Иерархия считается полной, если каждый элемент заданного уровня является критерием для всех элементов нижнего уровня



**Рисунок 1 – Полная доминантная иерархия.**

Критерии:

К 1 – Цена;

К 2 – Вес;

К 3 – Калорийность;

К 4 – Срок хранения;

Альтернативы:

А 1 - Простоквашино;

А 2 - Просто;

А 3 - Сарафаново;

А 4 – Зелёная линия;

А 5 – Искренне ваш.

## **1.3 Установка приоритетов критериев**

После иерархического представления задачи установлены приоритеты критериев и оценена каждая из альтернатив по критериям, определена наиболее важная их них. В методе анализа иерархий элементы сравниваются попарно по отношению к их влиянию на общую для них характеристику. Парные сравнения приводят к записи характеристик сравнений в виде квадратной таблицы чисел, которая называется матрицей. Для облегчения работы введена шкала относительной важности (Таблица 1).

*Таблица.1 – Шкала относительной важности.*

|  |  |  |
| --- | --- | --- |
| Интенсивность  относительной  важности | Определение | Объяснение |
| 1 | Равная важность | Равный вклад двух критериев в цель. |
| 3 | Слабое превосходство | Дают легкое превосходство одной альтернативы над другой |
| 5 | Умеренное превосходство | Опыт и суждения дают умеренное превосходство |
| 7 | Сильное превосходство | Одному из критериев дается настолько сильное предпочтение. |
| 9 | Абсолютное превосходство | Очевидность превосходства одного критерия над другим |
| 2,4,6,8 | Промежуточные решения между двумя соседними суждениями | Применяется в компромиссных случаях |

Шкала содержит соответствующие обратные значения.

## **1.4 Синтез приоритетов**

После построения иерархии и определения величин парных субъективных суждений следует этап, на котором иерархическая декомпозиция и относительные суждения объединяются для получения осмысленного решения многокритериальной задачи принятия решений. Из групп парных сравнений формируется набор локальных критериев, которые выражают относительное влияние элементов на элемент, расположенный на уровне выше. Составлена обратно симметричная матрица для парного сравнения критериев (Таблица 2).

*Таблица 2 – Матрица парного сравнения критериев.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Цель | К 1 | К 2 | К 3 | К 4 | Vi | W2i |
| К 1 | 1 | 1 | 3 | 5 | 1.968 | 0.391 |
| К 2 | 1 | 1 | 3 | 5 | 1.968 | 0.391 |
| К 3 | 1/3 | 1/3 | 1 | 3 | 0.760 | 0.151 |
| К 4 | 1/5 | 1/5 | 1/3 | 1 | 0.340 | 0.067 |
| ∑Vi | | | | | 5.036 |

Для определения относительной ценности каждого элемента необходимо найти геометрическое среднее и с этой целью перемножить n элементов каждой строки и из полученного результата извлечь корни n-й степени (размерность матрицы n=5).

Строка № 1

V1= (1х1х3х5)1/4 = 1.968;

Строка № 2

V2= (1х1х3х5)1/4 = 1.968;

Строка № 3

V3=(1/3x1/3x1x3)1/4 = 0.760;

Строка № 4

V4=(1/5x1/5x1/3x1)1/4 = 0.340;

Проведена нормализация полученных чисел. Для этого определен нормирующий коэффициент ∑Vi.

∑Vi = 1.968 + 1.968 + 0.760 + 0.340 = 5.036

Найдена важность приоритетов W2i, для этого каждое из чисел Vi разделено на ∑Vi.

***Строка № 1***

W21= 1.968 / 5.036 = 0.391;

Строка № 2

W22= 1.968 / 5.036 = 0.391;

***Строка № 3***

W23= 0.760 / 5.036 = 0.151;

***Строка № 4***

W24= 0.340 / 5.036 = 0.067;

В результате получен вектор приоритетов:

W2i = (0.391; 0.391; 0.151; 0.067), где индекс 2 означает, что вектор приоритетов относится ко второму уровню иерархии.

К 1 – Цена (Таблица 3);

*Таблица 3 – Матрица сравнения по критерию 1.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| К1 | А1 | А2 | А3 | А4 | А5 | VК1Y | W3К1Y |
| А1 | 1 | 1/5 | 1/7 | 1/7 | 1/9 | 0.214 | 0.028 |
| А2 | 5 | 1 | 1/5 | 1/5 | 1/7 | 0.491 | 0.065 |
| А3 | 7 | 5 | 1 | 1 | 1/3 | 1.635 | 0.217 |
| А4 | 7 | 5 | 1 | 1 | 1/3 | 1.635 | 0.217 |
| А5 | 9 | 7 | 3 | 3 | 1 | 3.554 | 0.472 |
| ∑VК1Y | | | | | | 7.529 |

Определена относительная ценность каждого элемента.

Строка № 1

VК11=(1x1/5x1/7x1/7x1/9)1/5= 0.214;

Строка № 2

VК12=(5x1x1/5x1/5x1/5x1/7)1/5= 0.491;

Строка № 3

VК13=(7x5x1x1x1/3)1/5= 1.635;

Строка № 4

VК14=(7x5x1x1x1/3)1/5= 1.635;

Строка № 5

VК15=(9x7x3x3x1)1/5= 3.554.

Проведена нормализация полученных чисел. Для этого определен нормирующий коэффициент ∑VK1Y.

∑VК1Y = 0.214 + 0.491 + 1.635 + 1.635 + 3.554 = 7.529.

Найдена важность приоритетов W3К1Y, для этого каждое из чисел VK1Y разделено на ∑VK1Y.

Строка № 1

W3К11= 0.214 / 7.529 = 0.028;

Строка № 2

W3К12= 0.491 / 7.529 = 0.065;

Строка № 3

W3К13= 1.635 / 7.529 = 0.217;

Строка № 4

W3К14= 1.635 / 7.529 = 0.217;

Строка № 5

W3К15= 3.554 / 7.529 = 0.472;

В результате получаем вектор приоритетов:

W3К1Y = (0.028; 0.065; 0.217; 0.217; 0.472)

где индекс 3 означает, что вектор приоритетов относится к третьему уровню иерархии критерия К1.

К 2 – Вес (Таблица 4);

*Таблица 4. – Матрица сравнения по критерию 2.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| К2 | А1 | А2 | А3 | А4 | А5 | VК2Y | W3К2Y |
| А1 | 1 | 5 | 1 | 5 | 7 | 2.809 | 0.389 |
| А2 | 1/5 | 1 | 1/5 | 1 | 3 | 0.654 | 0.091 |
| А3 | 1 | 5 | 1 | 5 | 7 | 2.809 | 0.389 |
| А4 | 1/5 | 1 | 1/5 | 1 | 3 | 0.654 | 0.091 |
| А5 | 1/7 | 1/3 | 1/7 | 1/3 | 1 | 0.296 | 0.041 |
| ∑VК2Y | | | | | | 7.223 |  |

Определена относительная ценность каждого элемента.

Строка № 1

VК21=(1x5x1x5x7)1/5= 2.809;

Строка № 2

VК22=(1/5x1x1/5x1x3)1/5=0.654;

Строка № 3

VК23=(1x5x1x5x7)1/5= 2.809;

Строка № 4

VК24=(1/5x1x1/5x1x3)1/5= 0.654;

Строка № 5

VК25=(1/7x1/3x1/7x1/3x1)1/5= 0.296.

Проведена нормализация полученных чисел. Для этого определен нормирующий коэффициент ∑VK2Y.

∑VК2Y = 2.809 + 0.654 + 2.809 + 0.654 + 0.296 = 7.223.

Найдена важность приоритетов W3К2Y, для этого каждое из чисел VK2Y разделено на ∑VK2Y.

Строка № 1

W3К21= 2.809 / 7.223 = 0.389;

Строка № 2

W3К22= 0.654 / 7.223 = 0.091;

Строка № 3

W3К23= 2.809 / 7.223 = 0.389;

Строка № 4

W3К24= 0.654 / 7.223 = 0.091;

Строка № 5

W3К25= 0.296 / 7.223 = 0.041.

В результате получаем вектор приоритетов:

W3К2Y = (0.389; 0.091; 0.389; 0.091; 0.041),

где индекс 3 означает, что вектор приоритетов относится к третьему уровню иерархии критерия К2.

К 3 – Калорийность (Таблица 5);

*Таблица 5 – Матрица сравнения по критерию 5.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| К3 | А1 | А2 | А3 | А4 | А5 | VК3Y | W3К3Y |
| А1 | 1 | 5 | 5 | 5 | 1/3 | 2.108 | 0.278 |
| А2 | 1/5 | 1 | 1 | 1 | 1/7 | 0.491 | 0.065 |
| А3 | 1/5 | 1 | 1 | 1 | 1/7 | 0.491 | 0.065 |
| А4 | 1/5 | 1 | 1 | 1 | 1/7 | 0.491 | 0.065 |
| А5 | 3 | 7 | 7 | 7 | 1 | 4.004 | 0.528 |
| VК35 | | | | | | 7.586 |

Определена относительная ценность каждого элемента.

Строка № 1

VК31= (1x5x5x5x1/3)1/5= 2.108;

Строка № 2

VК32= (1/5x1x1x1x1/7)1/5= 0.491;

Строка № 3

VК33= (1/5x1x1x1x1/7)1/5= 0.491;

Строка № 4

VК34= (1/5x1x1x1x1/7)1/5= 0.491;

Строка № 5

VК35= (3x7x7x7x1)1/5= 4.004.

Проведена нормализация полученных чисел. Для этого определен нормирующий коэффициент ∑VK3Y.

∑VК3Y = 2.108 + 0.491 + 0.491 + 0.491 + 4.004 = 7.586.

Найдена важность приоритетов W3К2Y, для этого каждое из чисел VK2Y разделено на ∑VK2Y.

Строка № 1

W3К31= 2.108 / 7.586 = 0.278;

Строка № 2

W3К32= 0.491 / 7.586 = 0.065;

Строка № 3

W3К33= 0.491 / 7.586 = 0.065;

Строка № 4

W3К34= 0.491 / 7.586 = 0.065;

Строка № 5

W3К35= 4.004 / 7.586 = 0.528;

В результате получаем вектор приоритетов:

W3К3Y = (0.278; 0.065; 0.065; 0.065; 0.528),

где индекс 3 означает, что вектор приоритетов относится к третьему уровню иерархии критерия К3.

К 4 – срок хранения (Таблица 6);

*Таблица 6 – Матрица сравнения по критерию 4.*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| К4 | А1 | А2 | А3 | А4 | А5 | VК4Y | W3К4Y |
| А1 | 1 | 9 | 1 | 9 | 9 | 3.737 | 0.423 |
| А2 | 1/9 | 1 | 1/9 | 1 | 5 | 0.573 | 0.065 |
| А3 | 1 | 9 | 1 | 9 | 9 | 3.737 | 0.423 |
| А4 | 1/9 | 1 | 1/9 | 1 | 5 | 0.573 | 0.065 |
| А5 | 1/9 | 1/5 | 1/9 | 1/5 | 1 | 0.218 | 0.025 |
| ∑VК4Y | | | | | | 8.838 |

Определена относительная ценность каждого элемента.

Строка № 1

VК41=(1x9x1x9x9)1/5 = 3.737;

Строка № 2

VК42=(1/9x1x1/9x1x5)1/5 = 0.573;

Строка № 3

VК43=(x9x1x9x9)1/5 = 3.737;

Строка № 4

VК44=(1/9x1x1/9x1x5)1/5 = 0.573;

Строка № 5

VК45=(1/9x1/5x1/9x1/5x1)1/5 = 0.218.

Проведена нормализация полученных чисел. Для этого определен нормирующий коэффициент ∑VK4Y.

∑VК4Y = 3.737 + 0.573 + 3.737 + 0.573 + 0.218 = 8.838.

Найдена важность приоритетов W3К4Y, для этого каждое из чисел VK4Y разделено на ∑VK4Y.

Строка № 1

W3К41= 3.737 / 8.838 = 0.423;

Строка № 2

W3К42= 0.573 / 8.838 = 0.065;

Строка № 3

W3К43= 3.737 / 8.838 = 0.423;

Строка № 4

W3К44= 0.573 / 8.838 = 0.065;

Строка № 5

W3К45= 0.218 / 8.838 = 0.025.

В результате получаем вектор приоритетов:

W3К4Y = (0.423; 0.065; 0.423; 0.065; 0.025),

где индекс 3 означает, что вектор приоритетов относится к третьему уровню иерархии критерия К4.

## **1.5 Согласованность локальных приоритетов**

Любая матрица суждений в общем случае не согласована, так как суждения отражают субъективные мнения ЛПР, а сравнение элементов, которые имеют количественные эквиваленты, может быть несогласованным из-за присутствия погрешности при проведении измерений. Совершенной согласованности парных сравнений даже в идеальном случае на практике достичь трудно. Нужен способ оценки степени согласованности при решении конкретной задачи.

Метод анализа иерархий дает возможность провести такую оценку.

Вместе с матрицей парных сравнений есть мера оценки степени отклонения от согласованности. Когда такие отклонения превышают установленные пределы тем, кто проводит решение задачи, необходимо их пересмотреть.

В таблице приведены средние значения индекса случайной согласованности (СИ) для случайных матриц суждений разного порядка.

В нашей задаче размерность матрицы n=5, тогда среднее значение индекса случайной согласованности СИ = 1,12.

Определены индекс согласованности и отношение согласованности для матрицы «Выбор лучшего отеля» (Таблица 8).

*Таблица 8 – Матрица «Выбор молока в магазине».*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Цель | К 1 | К 2 | К 3 | К 4 | W2i |
| К 1 | 1 | 1 | 3 | 5 | 0.391 |
| К 2 | 1 | 1 | 3 | 5 | 0.391 |
| К 3 | 1/3 | 1/3 | 1 | 3 | 0.151 |
| К 4 | 1/5 | 1/5 | 1/3 | 1 | 0.067 |

Определена сумма каждого столбца матрицы суждений.

S1 = 1 + 1 + 0.333 + 0.2 = 2.533;

S2 = 1 + 1 + 0.333 + 0.2 = 2.533;

S3 = 3 + 3 + 1 + 0.333 = 7.333;

S4 = 5 + 5 + 3 + 1 = 14;

Полученный результат умножен на компоненту нормализованного вектора приоритетов, т.е. сумму суждений первого столбца на первую компоненту, сумму суждений второго столбца - на вторую и т.д.

Р1 = S1 х W21 = 2.533 × 0.391 = 0.99;

Р2 = S2 х W22 = 2.533 × 0.391 = 0.99;

Р3 = S3 х W23 = 7.333 × 0.151 = 1.11;

Р4 = S4 х W24 = 14 × 0.067 = 0.94;

Сумма чисел Рj отражает пропорциональность предпочтений, чем ближе эта величина к n (числу объектов и видов действия в матрице парных сравнений), тем более согласованны суждения.

λmax = Р1 + Р2 + Р3 + Р4 = 4.03.

Отклонение от согласованности выражается индексом согласованности.

ИС = (λmax - n)/(n - 1) = (4.03 - 4)/(4 - 1) = 0.0105.

Отношение индекса согласованности ИС к среднему значению случайного индекса согласованности СИ называется отношением согласованности ОС.

ОС = ИС/СИ = 0.0105/0.9 = 0.012.

Значение ОС меньше или равное 0.10 считается приемлемым, значит матрица «Выбор лучшего спортсмена» согласована.

Определнены индекс согласованности и отношение согласованности для матрицы К 1 – цена (Таблица 9).

*Таблица 9 – Матрица сравнения по критерию 1.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| К1 | А1 | А2 | А3 | А4 | А5 | W3К1Y |
| А1 | 1 | 1/5 | 1/7 | 1/7 | 1/9 | 0.028 |
| А2 | 5 | 1 | 1/5 | 1/5 | 1/7 | 0.065 |
| А3 | 7 | 5 | 1 | 1 | 1/3 | 0.217 |
| А4 | 7 | 5 | 1 | 1 | 1/3 | 0.217 |
| А5 | 9 | 7 | 3 | 3 | 1 | 0.472 |

Определяется сумма каждого столбца матрицы суждений.

S1К1 = 1 + 5 + 7 + 7 + 9 = 29;

S2 К1 = 0.2 + 1 + 5 + 5 + 7 = 18.2;

S3 К1 = 0.143 + 0.2 + 1 + 1 + 3 = 5.343;

S4 К1 = 0.143 + 0.2 + 1 + 1 + 3 = 5.343;

S5 К1 =0.111 + 0.143 + 0.333 + 0.333 + 1 = 1.921.

Затем полученный результат умножен на компоненту нормализованного вектора приоритетов.

Р1 К1 = S1 х W3К11 = 29.000 × 0.028 = 0.83;

Р2 К1 = S2 х W3К12 = 18.200 × 0.065 = 1.19;

Р3 К1 = S3 х W3К13 = 5.343 × 0.217 = 1.16;

Р4 К1 = S1 х W3К14 = 5.343 × 0.217 = 1.16;

Р5 К1 = S1 х W3К15 = 1.921 × 0.472 = 0.91.

Найдена пропорциональность предпочтений.

λmax К1 = Р1К1 + Р2К1 + Р3К1 + Р4К1 + Р5К1 = 5.24.

Отклонение от согласованности выражается индексом согласованности.

ИС К1 = (λmax К1 - n)/(n - 1) = (5.24 - 5)/(5 - 1) = 0.0600.

Найдено отношение согласованности ОС.

ОС К1 = ИС/СИ = 0.0600/1.12 = 0.054.

Значение ОС меньше или равное 0.10 считается приемлемым, значит матрица К 1 (цена дня проживания) согласована.

Определены индекс согласованности и отношение согласованности для матрицы К 2 – вес (Таблица 10).

*Таблица 10 – Матрица сравнения по критерию 2.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| К2 | А1 | А2 | А3 | А4 | А5 | W3К2Y |
| А1 | 1 | 5 | 1 | 5 | 7 | 0.389 |
| А2 | 1/5 | 1 | 1/5 | 1 | 3 | 0.091 |
| А3 | 1 | 5 | 1 | 5 | 7 | 0.389 |
| А4 | 1/5 | 1 | 1/5 | 1 | 3 | 0.091 |
| А5 | 1/7 | 1/3 | 1/7 | 1/3 | 1 | 0.041 |

Определена сумма каждого столбца матрицы суждений.

S1К2 = 1 + 0.2 + 1 + 0.2 + 0.143 = 2.543;

S2 К2 = 5 + 1 + 5 + 1 + 0.333 = 12.333;

S3 К2 = 1 + 0.2 + 1 + 0.2 + 0.143 = 2.543;

S4 К2 = 5 + 1 + 5 + 1 + 0.333 = 12.333;

S5 К2 = 7 + 3 + 7 + 3 + 1 = 21.

Затем полученный результат умножен на компоненту нормализованного вектора приоритетов.

Р1 К2 = S1 х W3 К21 = 2.543 × 0.389 = 0.99;

Р2 К2 = S2 х W3 К22 = 12.333 × 0.091 = 1.12;

Р3 К2 = S3 х W3 К23 = 2.543 × 0.389 = 0.99;

Р4 К2 = S4 х W3 К24 = 12.333 × 0.091 = 1.12;

Р5 К2 = S5 х W3 К25 = 21.000 × 0.041 = 0.86.

Найдена пропорциональность предпочтений.

λmax К2 = Р1К2 + Р2К2 + Р3К2 + Р4К2 + Р5К2 = 5.07.

Отклонение от согласованности выражается индексом согласованности.

ИС К2 = (λmax К2 - n)/(n - 1) = (5.07 - 5)/(5 - 1) = 0.0182.

Найдено отношение согласованности ОС.

ОС К2 = ИС/СИ = 0.0182/1.12 = 0.016.

Значение ОС меньше или равное 0.10 считается приемлемым, значит матрица К 2 (количество звезд) согласована.

Определены индекс согласованности и отношение согласованности для матрицы К 3 – калорийность (Таблица 11).

*Таблица 11 – Матрица сравнения по критерию 3.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| К3 | А1 | А2 | А3 | А4 | А5 | W3К3Y |
| А1 | 1 | 5 | 5 | 5 | 1/3 | 0.278 |
| А2 | 1/5 | 1 | 1 | 1 | 1/7 | 0.065 |
| А3 | 1/5 | 1 | 1 | 1 | 1/7 | 0.065 |
| А4 | 1/5 | 1 | 1 | 1 | 1/7 | 0.065 |
| А5 | 3 | 7 | 7 | 7 | 1 | 0.528 |

Определена сумма каждого столбца матрицы суждений.

S1К3 = 1 + 0.2 + 0.2 + 0.2 + 3 = 4.600;

S2 К3 = 5 + 1 + 1 + 1 + 7 = 15;

S3 К3 = 5 + 1 + 1 + 1 + 7 = 15;

S4 К3 = 5 + 1 + 1 + 1 + 7 = 15;

S5 К3 = 0.333 + 0.143 + 0.143 + 0.143 + 1 = 1.762.

Затем полученный результат умножен на компоненту нормализованного вектора приоритетов.

Р1 К3 = S1 х W3 К31 = 4.6 × 0.278 = 1.28;

Р2 К3 = S2 х W3 К32 = 15 × 0.065 = 0.97;

Р3 К3 = S3 х W3 К33 = 15 × 0.065 = 0.97;

Р4 К3 = S4 х W3 К34 = 15 × 0.065 = 0.97;

Р5 К3 = S5 х W3 К35 = 1.762 × 0.528 = 0.93.

Найдем пропорциональность предпочтений.

λmax К3 = Р1К3 + Р2К3 + Р3К3 + Р4К3 + Р5К3 = 5.12.

Отклонение от согласованности выражается индексом согласованности.

ИС К3 = (λmax К3 - n)/(n - 1) = (5.12 - 5)/(5 - 1) = 0.0305.

Найдено отношение согласованности ОС.

ОС К3 = ИС/СИ = 0.0305/1.12 = 0.027.

Значение ОС меньше или равное 0.10 считается приемлемым, значит матрица К 3 (рейтинг по отзывам пользователей) согласована.

Определены индекс согласованности и отношение согласованности для матрицы К 4 – срок хранения (Таблица 12).

*Таблица 12 – Матрица сравнения по критерию 4.*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| К4 | А1 | А2 | А3 | А4 | А5 | W3К4Y |
| А1 | 1 | 9 | 1 | 9 | 9 | 0.423 |
| А2 | 1/9 | 1 | 1/9 | 1 | 5 | 0.065 |
| А3 | 1 | 9 | 1 | 9 | 9 | 0.423 |
| А4 | 1/9 | 1 | 1/9 | 1 | 5 | 0.065 |
| А5 | 1/9 | 1/5 | 1/9 | 1/5 | 1 | 0.025 |

Определена сумма каждого столбца матрицы суждений.

S1К4 = 1 + 0.111 + 1 + 0.111 + 0.111 = 2.333;

S2К4 = 9 + 1 + 9 + 1 + 0.2 = 20.2;

S3К4 = 1 + 0.111 + 1 + 0.111 + 0.111 = 2.333;

S4К4 = 9 + 1 + 9 + 1 + 0.2 = 20.2;

S5К4 = 9 + 5 + 9 + 5 + 1 = 29.

Затем полученный результат умножен на компоненту нормализованного вектора приоритетов.

Р1К4 = S1 х W3 К41 = 2.333 × 0.423 = 0.99;

Р2К4 = S2 х W3 К42 = 20.200 × 0.065 = 1.31;

Р3К4 = S3 х W3 К43 = 2.333 × 0.423 = 0.99;

Р4К4 = S4 х W3 К44 = 20.200 × 0.065 = 1.31;

Р5К4 = S5 х W3 К45 = 29.000 × 0.025 = 0.72.

Найдена пропорциональность предпочтений.

λmax К4 = Р1К4 + Р2К4 + Р3К4 + Р4К4 + Р5К4 = 5.31.

Отклонение от согласованности выражается индексом согласованности.

ИС К4 = (λmax К4 - n)/(n - 1) = (5.31 - 5)/(5 - 1) = 0.0769.

Найдено отношение согласованности ОС.

ОС К4 = ИС/СИ = 0.0769/1.12 = 0.069.

Значение ОС меньше или равное 0,10 считается приемлемым, значит матрица К 4 (удаленность от ближайшей станции метро) согласована.

## **3.6 Синтез альтернатив**

Векторы приоритетов и отношения согласованности определяются для всех матриц суждений, начиная со второго уровня.

Для определения приоритетов альтернатив локальные приоритеты умножены на приоритет соответствующего критерия на высшем уровне и найдены суммы по каждому элементу в соответствии с критериями, на которые воздействует этот элемент.

W2i = (0.391; 0.391; 0.151; 0.067);

W3К1Y = (0.028; 0.065; 0.217; 0.217; 0.472);

W3К2Y = (0.389; 0.091; 0.389; 0.091; 0.041);

W3К3Y = (0.278; 0.065; 0.065; 0.065; 0.528);

W3К4Y (0.423; 0.065; 0.423; 0.065; 0.025);

Приоритеты альтернатив получены следующим образом:

W1 = W21 х W3К11 + W22 х W3К21 + W23 х W3К31 + W24 х W3К41 = 0.391 x 0.028

+ 0.391 x 0.389 +0.151 x 0.278 + 0.067 x 0.423 = 0.233.

W2 = W21 х W3К12 + W22 х W3К22 + W23 х W3К32 + W24 х W3К42 = 0.391 × 0.065 + 0.391 × 0.091 + 0.151 × 0.065 + 0.067 × 0.065 = 0.075.

W3 = W21 х W3К13 + W22 х W3К23 + W23 х W3К33 + W24 х W3К43 = 0.391 × 0.217 + 0.391 × 0.389 + 0.151 × 0.065 + 0.067 × 0.423 = 0.275

W4 = W21 х W3К14 + W22 х W3К24 + W23 х W3К34 + W24 х W3К44 = 0.391 × 0.217 + 0.391 × 0.091 + 0.151 × 0.065 + 0.067 × 0.065 = 0.135.

W5 = W21 х W3К15 + W22 х W3К25 + W23 х W3К35 + W24 х W3К45 = 0.391 × 0.472 + 0.391 × 0.041 + 0.151 × 0.528 + 0.067 × 0.025 = 0.282.

Таким образом, приоритеты альтернатив равны:

альтернатива А1 (Сарафаново) - W1 приоритет равен = 0.233;

альтернатива А2 (Простоквашино)- W2 приоритет равен = 0.075;

альтернатива А3 (Просто) - W3 приоритет равен = 0.275;

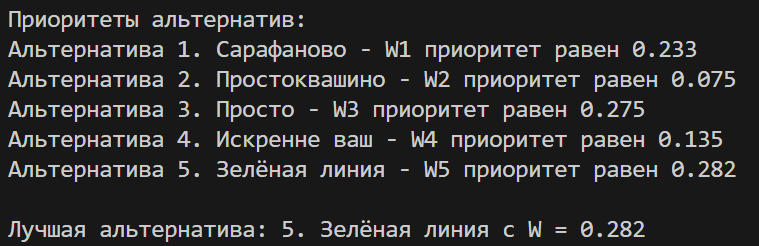
альтернатива А4 (Искренне ваш) – W4 приоритет равен = 0.135;

альтернатива А5 (Зелёная линия) - W5 приоритет равен = 0.282.

## **3.7 Вывод**

Лучшая альтернатива: 5. Зелёная линия с W = 0.282

## **1.8 Результаты работы программы**



**Рисунок 2 – Вывод программы**

**ЗАКЛЮЧЕНИЕ**

В ходе работы был использован метод анализа иерархий (МАИ) для решения задачи многокритериальной оценки и принятия решений. Метод был применён для сравнения альтернатив по различным критериям, что позволило выявить наиболее предпочтительные варианты. В результате применения МАИ были получены приоритеты альтернатив, что обеспечило объективное и структурированное решение.

Преимущества метода МАИ включают его способность систематизировать сложные задачи с множеством критериев, чётко формализовать процесс принятия решений и снизить влияние субъективных факторов. МАИ позволяет учитывать разные взгляды и предпочтения при принятии решения, что делает его универсальным инструментом для различных областей: от бизнеса до научных исследований.

Однако метод имеет и недостатки. Он может быть чувствителен к ошибкам в оценке сравнений, особенно когда различия между альтернативами незначительны. Кроме того, для корректной работы требуется точность и внимательность при составлении матрицы суждений, что может быть сложным для неопытных пользователей. Также важно учитывать, что при большом количестве критериев и альтернатив МАИ может стать вычислительно дорогим и трудоемким.

**СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1. Болотова Л. С. Многокритериальная оптимизация. Болотова Л. С., Сорокин А. Б. [Электронный ресурс] / Метод. указания по вып. курсовой работы — М.: МИРЭА, 2015.
2. Сорокин А. Б. Методы оптимизации: гибридные генетические алгоритмы. Сорокин А. Б. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2016.
3. Сорокин А. Б. Линейное программирование: практикум. Сорокин А. Б., Бражникова Е. В., Платонова О. В. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2017.

**ПРИЛОЖЕНИЯ**

Приложение А – Код реализации МАИ на языке python.

**Приложение А**

Код реализации МАИ на языке python.

Листинг А.1. Реализация МАИ.

import pandas as pd

# Установить формат отображения чисел

pd.options.display.float\_format = '{:.2f}'.format

# Альтернативы

alternatives = [

{'name': '1. Сарафаново', 'price': 129, 'weight': 970, 'calories': 46, 'shelf\_life': 180},

{'name': '2. Простоквашино', 'price': 99, 'weight': 930, 'calories': 53, 'shelf\_life': 16},

{'name': '3. Просто', 'price': 85, 'weight': 970, 'calories': 53, 'shelf\_life': 180},

{'name': '4. Искренне ваш', 'price': 85, 'weight': 930, 'calories': 53, 'shelf\_life': 16},

{'name': '5. Зелёная линия', 'price': 79, 'weight': 900, 'calories': 40, 'shelf\_life': 10},

]

df = pd.DataFrame(alternatives)

df.index = df.index + 1

print(df, '\n')

# Таблица критериев

criteria\_data = {

'К1': [1, 1, 1/3, 1/5],

'К2': [1, 1, 1/3, 1/5],

'К3': [3, 3, 1, 1/3],

'К4': [5, 5, 3, 1]

}

criteria\_df = pd.DataFrame(criteria\_data, index=['К1', 'К2', 'К3', 'К4'], dtype=float) # Преобразование в float

print("Таблица критериев:")

print(criteria\_df, '\n')

# Вычисление суммы каждой строки и вывод

print("Вычисление V:")

row\_products = criteria\_df.prod(axis=1) # Произведение элементов в строке

row\_roots = row\_products \*\* (1 / len(criteria\_df.columns)) # Корень степени n

for i, value in enumerate(row\_roots, start=1):

print(f"V{i} = ({'x'.join(map(str, criteria\_df.iloc[i-1]))})^(1/{len(criteria\_df.columns)}) = {value:.3f};")

# Вычисление суммы всех V и вывод W

sum\_v = row\_roots.sum()

print(f"ΣVi = {' + '.join(f'{v:.3f}' for v in row\_roots)} = {sum\_v:.3f}\n")

print("Вычисление W2i:")

for i, value in enumerate(row\_roots, start=1):

w = value / sum\_v

print(f"W2{i} = {value:.3f} / {sum\_v:.3f} = {w:.3f};")

# Вычисление W2i и вывод в формате (0.388; 0.388; 0.150; 0.075;)

w\_values = [f"{value / sum\_v:.3f}" for value in row\_roots]

print(f"W2i = ({'; '.join(w\_values)});\n")

# Таблица сравнения по критерию K1

k1\_data = {

'A1': [1, 1/5, 1/7, 1/7, 1/9],

'A2': [5, 1, 1/5, 1/5, 1/7],

'A3': [7, 5, 1, 1, 1/3],

'A4': [7, 5, 1, 1, 1/3],

'A5': [9, 7, 3, 3, 1]

}

k1\_df = pd.DataFrame.from\_dict(k1\_data, orient='index', dtype=float, columns=['A1', 'A2', 'A3', 'A4', 'A5'])

print("Таблица сравнения по критерию K1:")

print(k1\_df, '\n')

# Вычисление V\_{K1i}

print("Вычисление V\_K1i:")

k1\_row\_products = k1\_df.prod(axis=1) # Произведение элементов в строке

k1\_row\_roots = k1\_row\_products \*\* (1 / len(k1\_df.columns)) # Корень степени n

for i, value in enumerate(k1\_row\_roots, start=1):

print(f"V\_K1{i} = ({'x'.join(map(str, k1\_df.iloc[i-1]))})^(1/{len(k1\_df.columns)}) = {value:.3f};")

# Вычисление суммы всех V\_{K1i} и вывод W\_{K1i}

k1\_sum\_v = k1\_row\_roots.sum()

print(f"ΣV\_K1i = {' + '.join(f'{v:.3f}' for v in k1\_row\_roots)} = {k1\_sum\_v:.3f}\n")

print("Вычисление W\_K1i:")

for i, value in enumerate(k1\_row\_roots, start=1):

w = value / k1\_sum\_v

print(f"W\_K1{i} = {value:.3f} / {k1\_sum\_v:.3f} = {w:.3f};")

# Вывод W\_{K1i} в формате (0.388; 0.388; 0.150; 0.075;)

k1\_w\_values = [f"{value / k1\_sum\_v:.3f}" for value in k1\_row\_roots]

print(f"W\_K1i = ({'; '.join(k1\_w\_values)});\n")

# Таблица сравнения по критерию K2

k2\_data = {

'A1': [1, 5, 1, 5, 7],

'A2': [1/5, 1, 1/5, 1, 3],

'A3': [1, 5, 1, 5, 7],

'A4': [1/5, 1, 1/5, 1, 3],

'A5': [1/7, 1/3, 1/7, 1/3, 1]

}

k2\_df = pd.DataFrame.from\_dict(k2\_data, orient='index', dtype=float, columns=['A1', 'A2', 'A3', 'A4', 'A5'])

print("Таблица сравнения по критерию K2:")

print(k2\_df, '\n')

# Вычисление V\_{K2i}

print("Вычисление V\_K2i:")

k2\_row\_products = k2\_df.prod(axis=1) # Произведение элементов в строке

k2\_row\_roots = k2\_row\_products \*\* (1 / len(k2\_df.columns)) # Корень степени n

for i, value in enumerate(k2\_row\_roots, start=1):

print(f"V\_K2{i} = ({'x'.join(map(str, k2\_df.iloc[i-1]))})^(1/{len(k2\_df.columns)}) = {value:.3f};")

# Вычисление суммы всех V\_{K2i} и вывод W\_{K2i}

k2\_sum\_v = k2\_row\_roots.sum()

print(f"ΣV\_K2i = {' + '.join(f'{v:.3f}' for v in k2\_row\_roots)} = {k2\_sum\_v:.3f}\n")

print("Вычисление W\_K2i:")

for i, value in enumerate(k2\_row\_roots, start=1):

w = value / k2\_sum\_v

print(f"W\_K2{i} = {value:.3f} / {k2\_sum\_v:.3f} = {w:.3f};")

# Вывод W\_{K2i} в формате (0.388; 0.388; 0.150; 0.075;)

k2\_w\_values = [f"{value / k2\_sum\_v:.3f}" for value in k2\_row\_roots]

print(f"W\_K2i = ({'; '.join(k2\_w\_values)});\n")

# Таблица сравнения по критерию K3

k3\_data = {

'A1': [1, 5, 5, 5, 1/3],

'A2': [1/5, 1, 1, 1, 1/7],

'A3': [1/5, 1, 1, 1, 1/7],

'A4': [1/5, 1, 1, 1, 1/7],

'A5': [3, 7, 7, 7, 1]

}

k3\_df = pd.DataFrame.from\_dict(k3\_data, orient='index', dtype=float, columns=['A1', 'A2', 'A3', 'A4', 'A5'])

print("Таблица сравнения по критерию K3:")

print(k3\_df, '\n')

# Вычисление V\_{K3i}

print("Вычисление V\_K3i:")

k3\_row\_products = k3\_df.prod(axis=1) # Произведение элементов в строке

k3\_row\_roots = k3\_row\_products \*\* (1 / len(k3\_df.columns)) # Корень степени n

for i, value in enumerate(k3\_row\_roots, start=1):

print(f"V\_K3{i} = ({'x'.join(map(str, k3\_df.iloc[i-1]))})^(1/{len(k3\_df.columns)}) = {value:.3f};")

# Вычисление суммы всех V\_{K3i} и вывод W\_{K3i}

k3\_sum\_v = k3\_row\_roots.sum()

print(f"ΣV\_K3i = {' + '.join(f'{v:.3f}' for v in k3\_row\_roots)} = {k3\_sum\_v:.3f}\n")

print("Вычисление W\_K3i:")

for i, value in enumerate(k3\_row\_roots, start=1):

w = value / k3\_sum\_v

print(f"W\_K3{i} = {value:.3f} / {k3\_sum\_v:.3f} = {w:.3f};")

# Вывод W\_{K3i} в формате (0.388; 0.388; 0.150; 0.075;)

k3\_w\_values = [f"{value / k3\_sum\_v:.3f}" for value in k3\_row\_roots]

print(f"W\_K3i = ({'; '.join(k3\_w\_values)});\n")

# Таблица сравнения по критерию K4

k4\_data = {

'A1': [1, 9, 1, 9, 9],

'A2': [1/9, 1, 1/9, 1, 5],

'A3': [1, 9, 1, 9, 9],

'A4': [1/9, 1, 1/9, 1, 5],

'A5': [1/9, 1/5, 1/9, 1/5, 1]

}

k4\_df = pd.DataFrame.from\_dict(k4\_data, orient='index', dtype=float, columns=['A1', 'A2', 'A3', 'A4', 'A5'])

print("Таблица сравнения по критерию K4:")

print(k4\_df, '\n')

# Вычисление V\_{K4i}

print("Вычисление V\_K4i:")

k4\_row\_products = k4\_df.prod(axis=1) # Произведение элементов в строке

k4\_row\_roots = k4\_row\_products \*\* (1 / len(k4\_df.columns)) # Корень степени n

for i, value in enumerate(k4\_row\_roots, start=1):

print(f"V\_K4{i} = ({'x'.join(map(str, k4\_df.iloc[i-1]))})^(1/{len(k4\_df.columns)}) = {value:.3f};")

# Вычисление суммы всех V\_{K4i} и вывод W\_{K4i}

k4\_sum\_v = k4\_row\_roots.sum()

print(f"ΣV\_K4i = {' + '.join(f'{v:.3f}' for v in k4\_row\_roots)} = {k4\_sum\_v:.3f}\n")

print("Вычисление W\_K4i:")

for i, value in enumerate(k4\_row\_roots, start=1):

w = value / k4\_sum\_v

print(f"W\_K4{i} = {value:.3f} / {k4\_sum\_v:.3f} = {w:.3f};")

# Вывод W\_{K4i}

k4\_w\_values = [f"{value / k4\_sum\_v:.3f}" for value in k4\_row\_roots]

print(f"W\_K4i = ({'; '.join(k4\_w\_values)});\n")

# Вычисление и вывод сумм столбцов (Si) для таблицы критериев

print("Вычисление сумм столбцов критериев:")

for i, col in enumerate(criteria\_df.columns, start=1):

column\_sum = criteria\_df[col].sum()

elements = ' + '.join([f"{x:.3f}".rstrip('0').rstrip('.') if isinstance(x, float) else str(x) for x in criteria\_df[col]])

print(f"S\_{i} = {elements} = {column\_sum:.3f};")

# ... (предыдущий код остается без изменений до вычисления W2i)

# Вычисление и вывод P\_i = S\_i \* W2i

print("\nВычисление P\_i:")

for i, (s, w) in enumerate(zip(criteria\_df.sum(), row\_roots / sum\_v), start=1):

p = s \* w

print(f"P\_{i} = S\_{i} × W2{i} = {s:.3f} × {w:.3f} = {p:.2f};")

# Вычисление λ\_max, ИС и ОС

n = 4 # Количество критериев

SI = 0.90 # Средний индекс согласованности для n=4

# Сумма P\_i (λ\_max)

lambda\_max = sum([s \* w for s, w in zip(criteria\_df.sum(), row\_roots / sum\_v)])

print(f"\nλ\_max = P\_1 + P\_2 + P\_3 + P\_4 = {lambda\_max:.2f}.")

# Индекс согласованности (ИС)

IC = (lambda\_max - n) / (n - 1)

print(f"ИС = (λ\_max - n)/(n - 1) = ({lambda\_max:.2f} - {n})/({n} - 1) = {IC:.4f}.")

# Отношение согласованности (ОС)

OC = IC / SI

print(f"ОС = ИС/СИ = {IC:.4f}/{SI} = {OC:.3f}.")

# Проверка согласованности

if OC <= 0.10:

print("Значение ОС меньше или равно 0.10 → матрица согласована.")

else:

print("Значение ОС превышает 0.10 → требуется пересмотр матрицы.")

# ... (после вывода W\_K1i)

# --- Расчёты для K1 ---

n\_k1 = 5 # Количество альтернатив

SI\_k1 = 1.12 # СИ для n=5

# 1. Суммы столбцов Sᵢ для K1

print("\nВычисление сумм столбцов для K1:")

S\_k1 = k1\_df.sum()

for i, s in enumerate(S\_k1, start=1):

print(f"S\_K1\_{i} = {' + '.join(f'{x:.3f}'.rstrip('0').rstrip('.') for x in k1\_df[f'A{i}'])} = {s:.3f};")

# 2. Pᵢ = Sᵢ × W\_K1i

print("\nВычисление P\_K1i:")

P\_k1 = S\_k1 \* (k1\_row\_roots / k1\_sum\_v)

for i, (s, w, p) in enumerate(zip(S\_k1, k1\_row\_roots / k1\_sum\_v, P\_k1), start=1):

print(f"P\_K1\_{i} = S\_K1\_{i} × W\_K1\_{i} = {s:.3f} × {w:.3f} = {p:.2f};")

# 3. λ\_max, ИС, ОС

lambda\_max\_k1 = P\_k1.sum()

IC\_k1 = (lambda\_max\_k1 - n\_k1) / (n\_k1 - 1)

OC\_k1 = IC\_k1 / SI\_k1

print(f"\nλ\_max\_K1 = P\_K1\_1 + P\_K1\_2 + P\_K1\_3 + P\_K1\_4 + P\_K1\_5 = {lambda\_max\_k1:.2f}.")

print(f"ИС\_K1 = (λ\_max\_K1 - n)/(n - 1) = ({lambda\_max\_k1:.2f} - {n\_k1})/({n\_k1} - 1) = {IC\_k1:.4f}.")

print(f"ОС\_K1 = ИС\_K1/СИ = {IC\_k1:.4f}/{SI\_k1} = {OC\_k1:.3f}.")

if OC\_k1 <= 0.10:

print("ОС\_K1 ≤ 0.10 → матрица K1 согласована.")

else:

print("ОС\_K1 > 0.10 → требуется пересмотр матрицы K1.")

# ... (после вывода W\_K2i)

# --- Расчёты для K2 ---

n\_k2 = 5 # Количество альтернатив

SI\_k2 = 1.12 # СИ для n=5

# 1. Суммы столбцов Sᵢ для K2

print("\nВычисление сумм столбцов для K2:")

S\_k2 = k2\_df.sum()

for i, s in enumerate(S\_k2, start=1):

print(f"S\_K2\_{i} = {' + '.join(f'{x:.3f}'.rstrip('0').rstrip('.') for x in k2\_df[f'A{i}'])} = {s:.3f};")

# 2. Pᵢ = Sᵢ × W\_K2i

print("\nВычисление P\_K2i:")

P\_k2 = S\_k2 \* (k2\_row\_roots / k2\_sum\_v)

for i, (s, w, p) in enumerate(zip(S\_k2, k2\_row\_roots / k2\_sum\_v, P\_k2), start=1):

print(f"P\_K2\_{i} = S\_K2\_{i} × W\_K2\_{i} = {s:.3f} × {w:.3f} = {p:.2f};")

# 3. λ\_max, ИС, ОС

lambda\_max\_k2 = P\_k2.sum()

IC\_k2 = (lambda\_max\_k2 - n\_k2) / (n\_k2 - 1)

OC\_k2 = IC\_k2 / SI\_k2

print(f"\nλ\_max\_K2 = P\_K2\_1 + P\_K2\_2 + P\_K2\_3 + P\_K2\_4 + P\_K2\_5 = {lambda\_max\_k2:.2f}.")

print(f"ИС\_K2 = (λ\_max\_K2 - n)/(n - 1) = ({lambda\_max\_k2:.2f} - {n\_k2})/({n\_k2} - 1) = {IC\_k2:.4f}.")

print(f"ОС\_K2 = ИС\_K2/СИ = {IC\_k2:.4f}/{SI\_k2} = {OC\_k2:.3f}.")

if OC\_k2 <= 0.10:

print("ОС\_K2 ≤ 0.10 → матрица K2 согласована.")

else:

print("ОС\_K2 > 0.10 → требуется пересмотр матрицы K2.")

# --- Расчёты для K3 ---

n\_k3 = 5 # Количество альтернатив

SI\_k3 = 1.12 # СИ для n=5

# 1. Суммы столбцов Sᵢ для K3

print("\nВычисление сумм столбцов для K3:")

S\_k3 = k3\_df.sum()

for i, s in enumerate(S\_k3, start=1):

print(f"S\_K3\_{i} = {' + '.join(f'{x:.3f}'.rstrip('0').rstrip('.') for x in k3\_df[f'A{i}'])} = {s:.3f};")

# 2. Pᵢ = Sᵢ × W\_K3i

print("\nВычисление P\_K3i:")

P\_k3 = S\_k3 \* (k3\_row\_roots / k3\_sum\_v)

for i, (s, w, p) in enumerate(zip(S\_k3, k3\_row\_roots / k3\_sum\_v, P\_k3), start=1):

print(f"P\_K3\_{i} = S\_K3\_{i} × W\_K3\_{i} = {s:.3f} × {w:.3f} = {p:.2f};")

# 3. λ\_max, ИС, ОС

lambda\_max\_k3 = P\_k3.sum()

IC\_k3 = (lambda\_max\_k3 - n\_k3) / (n\_k3 - 1)

OC\_k3 = IC\_k3 / SI\_k3

print(f"\nλ\_max\_K3 = P\_K3\_1 + P\_K3\_2 + P\_K3\_3 + P\_K3\_4 + P\_K3\_5 = {lambda\_max\_k3:.2f}.")

print(f"ИС\_K3 = (λ\_max\_K3 - n)/(n - 1) = ({lambda\_max\_k3:.2f} - {n\_k3})/({n\_k3} - 1) = {IC\_k3:.4f}.")

print(f"ОС\_K3 = ИС\_K3/СИ = {IC\_k3:.4f}/{SI\_k3} = {OC\_k3:.3f}.")

if OC\_k3 <= 0.10:

print("ОС\_K3 ≤ 0.10 → матрица K3 согласована.")

else:

print("ОС\_K3 > 0.10 → требуется пересмотр матрицы K3.")

# --- Расчёты для K4 ---

n\_k4 = 5 # Количество альтернатив

SI\_k4 = 1.12 # СИ для n=5

# 1. Суммы столбцов Sᵢ для K4

print("\nВычисление сумм столбцов для K4:")

S\_k4 = k4\_df.sum()

for i, s in enumerate(S\_k4, start=1):

print(f"S\_K4\_{i} = {' + '.join(f'{x:.3f}'.rstrip('0').rstrip('.') for x in k4\_df[f'A{i}'])} = {s:.3f};")

# 2. Pᵢ = Sᵢ × W\_K4i

print("\nВычисление P\_K4i:")

P\_k4 = S\_k4 \* (k4\_row\_roots / k4\_sum\_v)

for i, (s, w, p) in enumerate(zip(S\_k4, k4\_row\_roots / k4\_sum\_v, P\_k4), start=1):

print(f"P\_K4\_{i} = S\_K4\_{i} × W\_K4\_{i} = {s:.3f} × {w:.3f} = {p:.2f};")

# 3. λ\_max, ИС, ОС

lambda\_max\_k4 = P\_k4.sum()

IC\_k4 = (lambda\_max\_k4 - n\_k4) / (n\_k4 - 1)

OC\_k4 = IC\_k4 / SI\_k4

print(f"\nλ\_max\_K4 = P\_K4\_1 + P\_K4\_2 + P\_K4\_3 + P\_K4\_4 + P\_K4\_5 = {lambda\_max\_k4:.2f}.")

print(f"ИС\_K4 = (λ\_max\_K4 - n)/(n - 1) = ({lambda\_max\_k4:.2f} - {n\_k4})/({n\_k4} - 1) = {IC\_k4:.4f}.")

print(f"ОС\_K4 = ИС\_K4/СИ = {IC\_k4:.4f}/{SI\_k4} = {OC\_k4:.3f}.")

if OC\_k4 <= 0.10:

print("ОС\_K4 ≤ 0.10 → матрица K4 согласована.")

else:

print("ОС\_K4 > 0.10 → требуется пересмотр матрицы K4.")

# Вывод всех W

print("\nИтоговые значения W:")

print(f"W2i = ({'; '.join(w\_values)});")

print(f"W3K1Y = ({'; '.join(k1\_w\_values)});")

print(f"W3K2Y = ({'; '.join(k2\_w\_values)});")

print(f"W3K3Y = ({'; '.join(k3\_w\_values)});")

print(f"W3K4Y = ({'; '.join(k4\_w\_values)});")

# Рассчет W1, W2, W3, W4, W5

print("\nРассчет W1, W2, W3, W4, W5:")

# Преобразование строковых значений W в числовые

W2i = list(map(float, w\_values))

W3K1Y = list(map(float, k1\_w\_values))

W3K2Y = list(map(float, k2\_w\_values))

W3K3Y = list(map(float, k3\_w\_values))

W3K4Y = list(map(float, k4\_w\_values))

# Формулы для W1, W2, W3, W4, W5

W1 = sum(W2i[j] \* W for j, W in enumerate([W3K1Y[0], W3K2Y[0], W3K3Y[0], W3K4Y[0]]))

W2 = sum(W2i[j] \* W for j, W in enumerate([W3K1Y[1], W3K2Y[1], W3K3Y[1], W3K4Y[1]]))

W3 = sum(W2i[j] \* W for j, W in enumerate([W3K1Y[2], W3K2Y[2], W3K3Y[2], W3K4Y[2]]))

W4 = sum(W2i[j] \* W for j, W in enumerate([W3K1Y[3], W3K2Y[3], W3K3Y[3], W3K4Y[3]]))

W5 = sum(W2i[j] \* W for j, W in enumerate([W3K1Y[4], W3K2Y[4], W3K3Y[4], W3K4Y[4]]))

# Вывод результатов

print(f"W1 = {' + '.join([f'{W2i[j]:.3f} × {W:.3f}' for j, W in enumerate([W3K1Y[0], W3K2Y[0], W3K3Y[0], W3K4Y[0]])])} = {W1:.3f}")

print(f"W2 = {' + '.join([f'{W2i[j]:.3f} × {W:.3f}' for j, W in enumerate([W3K1Y[1], W3K2Y[1], W3K3Y[1], W3K4Y[1]])])} = {W2:.3f}")

print(f"W3 = {' + '.join([f'{W2i[j]:.3f} × {W:.3f}' for j, W in enumerate([W3K1Y[2], W3K2Y[2], W3K3Y[2], W3K4Y[2]])])} = {W3:.3f}")

print(f"W4 = {' + '.join([f'{W2i[j]:.3f} × {W:.3f}' for j, W in enumerate([W3K1Y[3], W3K2Y[3], W3K3Y[3], W3K4Y[3]])])} = {W4:.3f}")

print(f"W5 = {' + '.join([f'{W2i[j]:.3f} × {W:.3f}' for j, W in enumerate([W3K1Y[4], W3K2Y[4], W3K3Y[4], W3K4Y[4]])])} = {W5:.3f}")

# Приоритеты альтернатив

print("\nПриоритеты альтернатив:")

alternatives\_names = [alt['name'] for alt in alternatives]

for i, W in enumerate([W1, W2, W3, W4, W5], start=1):

print(f"Альтернатива {alternatives\_names[i-1]} - W{i} приоритет равен {W:.3f}")

# Определение лучшей альтернативы

best\_index = max(range(len([W1, W2, W3, W4, W5])), key=lambda i: [W1, W2, W3, W4, W5][i])

best\_alternative = alternatives\_names[best\_index]

print(f"\nЛучшая альтернатива: {best\_alternative} с W = {[W1, W2, W3, W4, W5][best\_index]:.3f}")