

4 ЛАБОРАТОРНАЯ РАБОТА №4

«МЕТОД АНАЛИЗА ИЕРАРХИЙ»

4.1 Цель работы

Углубление теоретических знаний в области системного анализа, приобретение навыков создания и описания иерархических структур, а также изучение понятий цель, критерий, альтернатива.

4.2 Вариант задания – 16 (4)

Требуется решить задачу методом анализа иерархий вручную, а затем написать программу, которая решает МАИ задачу любой размерности.

Фирме нужно купить оборудование для очистки воды помещений. Нужно выбрать из возможных альтернатив (табл. 4.1).

Таблица 4.1 – Критерии альтернатив для оборудования для очистки воды

Критерии	А	В	С
Стоимость	100	80	50
Степень очистки	98	95	90
Надежность	85	85	70
Производительность	0.2	0.1	0.05
Долговечность	4	5	5

4.3 Ход работы

4.3.1 Для начала с помощью метода анализа иерархий в ручную были проведены исследования в ходе которых было выявлено, что вариант “А” получает наивысший вес и следовательно является наиболее оптимальным. Все вычисления представлены на рисунках 4.1-4.6.

Метод анализа иерархий Вариант-4

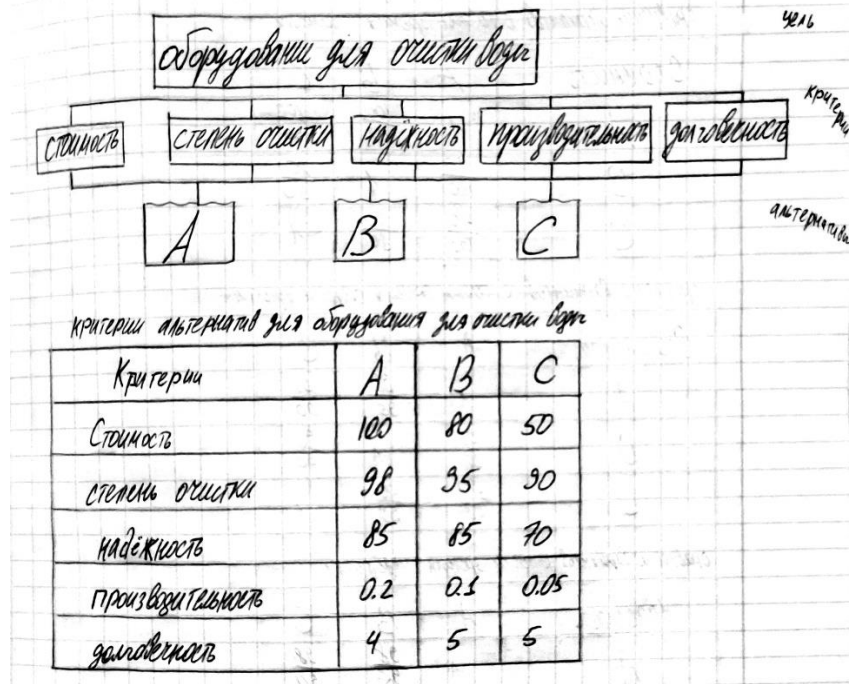


Рисунок 1.1 – Первая часть ручных вычислений

Составим матрицы попарных сравнений для пункта 3.

сравнение вариантов с точки зрения стоимости

Стоимость	А	В	С
А	1	$\frac{100}{80}$	$\frac{100}{50}$
В	$\frac{80}{100}$	1	$\frac{80}{50}$
С	$\frac{50}{100}$	$\frac{50}{80}$	1

сравнение вариантов с точки зрения степени очистки

Степень очистки	А	В	С
А	1	$\frac{98}{95}$	$\frac{98}{90}$
В	$\frac{95}{98}$	1	$\frac{95}{90}$
С	$\frac{90}{98}$	$\frac{90}{95}$	1

сравнение вариантов с точки зрения надежности

Надежность	А	В	С
А	1	1	$\frac{85}{70}$
В	1	1	$\frac{85}{70}$
С	$\frac{70}{85}$	$\frac{70}{85}$	1

Рисунок 1.2 – Вторая часть ручных вычислений

сравниваем варианты с точки зрения производительности

Производительность	A	B	C
A	1	$\frac{0.2}{0.1}$	$\frac{0.2}{0.05}$
B	$\frac{0.1}{0.2}$	1	$\frac{0.1}{0.05}$
C	$\frac{0.05}{0.2}$	$\frac{0.05}{0.1}$	1

сравниваем варианты с точки зрения долговечности

Долговечность	A	B	C
A	1	$\frac{4}{5}$	$\frac{4}{5}$
B	$\frac{5}{4}$	1	1
C	$\frac{5}{4}$	1	1

рассчитаем векторы приоритетов:

$$W_{11} = \sqrt[3]{1 \cdot \frac{100}{80} \cdot \frac{100}{50}} = \sqrt[3]{2.5} = 1.357$$

$$W_{12} = \sqrt[3]{\frac{80}{100} \cdot 1 \cdot \frac{80}{50}} = \sqrt[3]{1.28} = 1.086$$

$$W_{13} = \sqrt[3]{\frac{50}{100} \cdot \frac{50}{80} \cdot 1} = \sqrt[3]{0.3125} = 0.679$$

$$W_1 = 1.357 + 1.086 + 0.679 = 3.122$$

$$W_{21} = \sqrt[3]{1 \cdot \frac{95}{95} \cdot \frac{95}{30}} = \sqrt[3]{1.123} = 1.039$$

$$W_{22} = \sqrt[3]{\frac{95}{95} \cdot 1 \cdot \frac{95}{30}} = \sqrt[3]{1.013} = 1.008$$

$$W_{23} = \sqrt[3]{\frac{30}{95} \cdot \frac{30}{95} \cdot 1} = \sqrt[3]{0.87} = 0.954$$

Рисунок 1.3 – Третья часть ручных вычислений

$$W_2 = 1.039 + 1.008 + 0.954 = 3.001$$

$$W_{31} = \sqrt[3]{1 \cdot 1 \cdot \frac{85}{70}} = \sqrt[3]{1.214} = 1.067$$

$$W_{32} = \sqrt[3]{1 \cdot 1 \cdot \frac{85}{70}} = \sqrt[3]{1.214} = 1.067$$

$$W_{33} = \sqrt[3]{\frac{70}{85} \cdot \frac{70}{85} \cdot 1} = \sqrt[3]{\frac{0.823}{1.644}} = \frac{1}{1.644} = 0.618$$

$$W_3 = 1.067 + 1.067 + 0.618 = 3.314$$

$$W_{41} = \sqrt[3]{1 \cdot \frac{0.2}{0.1} \cdot \frac{0.2}{0.05}} = \sqrt[3]{8} = 2$$

$$W_{42} = \sqrt[3]{\frac{0.1}{0.2} \cdot 1 \cdot \frac{0.1}{0.05}} = \sqrt[3]{1} = 1$$

$$W_{43} = \sqrt[3]{\frac{0.05}{0.2} \cdot \frac{0.05}{0.1} \cdot 1} = \sqrt[3]{0.125} = 0.5$$

$$W_4 = 2 + 1 + 0.5 = 3.5$$

$$W_{51} = \sqrt[3]{1 \cdot \frac{4}{5} \cdot \frac{4}{5}} = \sqrt[3]{0.64} = 0.862$$

$$W_{52} = \sqrt[3]{\frac{5}{4} \cdot 1 \cdot 1} = \sqrt[3]{1.25} = 1.077$$

$$W_{53} = \sqrt[3]{\frac{5}{4} \cdot 1 \cdot 1} = \sqrt[3]{1.25} = 1.077$$

$$W_5 = 0.862 + 1.077 + 1.077 = 3.016$$

Рисунок 1.4 – Четвертая часть ручных вычислений

вариант решения	Стоимость	стеречь отходы	Надёжность	производительность	долговечность
A	$\frac{1,357}{3,122} = 0,435$	$\frac{1,039}{3,011} = 0,346$	$\frac{1,067}{3,314} = 0,322$	$\frac{2}{3,5} = 0,571$	$\frac{1,077}{3,324} = 0,324$
B	$\frac{1,088}{3,122} = 0,348$	$\frac{1,001}{3,011} = 0,336$	$\frac{1,067}{3,314} = 0,322$	$\frac{1}{3,5} = 0,286$	$\frac{1,077}{3,324} = 0,324$
C	$\frac{0,679}{3,122} = 0,217$	$\frac{0,954}{3,011} = 0,318$	$\frac{0,983}{3,314} = 0,297$	$\frac{0,5}{3,5} = 0,143$	$\frac{1,077}{3,324} = 0,324$

вектора приоритетов →

составим матрицу парных сравнений а затем сформируем вектор приоритетов. на основе экспертных оценок

образы удовлетворения образованию	1	2	3	4	5
Стоимость (1)	1	1	$\frac{1}{6}$	$\frac{1}{3}$	2
стеречь отходы (2)	1	1	5	8	$\frac{1}{2}$
Надёжность (3)	6	$\frac{1}{5}$	1	6	$\frac{1}{4}$
производительность (4)	3	$\frac{1}{8}$	$\frac{1}{6}$	1	3
Долговечность (5)	$\frac{1}{2}$	2	4	$\frac{1}{3}$	1

$$W_1 = \sqrt[5]{1 \cdot 1 \cdot \frac{1}{6} \cdot \frac{1}{3} \cdot 2} = 0,644$$

$$W_2 = \sqrt[5]{1 \cdot 1 \cdot 5 \cdot \frac{1}{8} \cdot \frac{1}{2}} = 1,820$$

$$W_3 = \sqrt[5]{6 \cdot \frac{1}{5} \cdot 1 \cdot 6 \cdot \frac{1}{4}} = 1,125$$

$$W_4 = \sqrt[5]{3 \cdot \frac{1}{8} \cdot \frac{1}{6} \cdot 1 \cdot 3} = 0,715$$

$$W_5 = \sqrt[5]{\frac{1}{2} \cdot 2 \cdot 4 \cdot \frac{1}{3} \cdot 1} = 1,059$$

$$W = 0,644 + 1,820 + 1,125 + 0,715 + 1,059 = 5,363$$

~~вектор приоритетов~~
~~нормализация~~

Рисунок 1.5 – Пятая часть ручных вычислений

Стоимость	стеречь отходы	Надёжность	производительность	Долговечность
$\frac{0,644}{5,363} = 0,120$	$\frac{1,820}{5,363} = 0,339$	$\frac{1,125}{5,363} = 0,210$	$\frac{0,715}{5,363} = 0,133$	$\frac{1,059}{5,363} = 0,197$

Вычислим итоговые приоритеты:

$$K_A = 0,435 \cdot 0,12 + 0,346 \cdot 0,339 + 0,322 \cdot 0,21 + 0,571 \cdot 0,133 + 0,324 \cdot 0,197 = 0,312$$

$$K_B = 0,348 \cdot 0,12 + 0,336 \cdot 0,339 + 0,322 \cdot 0,21 + 0,286 \cdot 0,133 + 0,324 \cdot 0,197 = 0,325$$

$$K_C = 0,217 \cdot 0,12 + 0,318 \cdot 0,339 + 0,297 \cdot 0,21 + 0,143 \cdot 0,133 + 0,324 \cdot 0,197 = 0,288$$

Вывод: На основе проведённых вычислений вариант "А" получает наименьший вес и => является наиболее оптимальным.

Рисунок 1.6 – Шестая часть ручных вычислений

4.3.2 Так же была создана программа на языке C++, вычисляющая МАИ любой размерности:

```
#include <iostream>

#include <windows.h>

#include <cmath>

class Matrix {

    double **mainMatrix; // критерии альтернатив

    double **matrixSravneniy; // матрица попарных сравнений на основе экспертных оценок

    // количество критериев и количество альтернатив

    int kolKrit, kolAlter; // строки = kolKrit; столбцы = kolAlter;

    //можно ввести 15 имен длиной в 19 символов

    char nameKrit[15][20]; // имена критериев

    char nameAlter[26]; // имена альтернатив

    char nameCeli[255]; // имя цели

public:

    Matrix();

    ~Matrix();

    void zapolniaem();

    void reshaem();

};

Matrix :: Matrix() {

    std::cout << "Введите количество критериев (kol <= 15): ";

    std::cin >> this->kolKrit;

    std::cout << "Введите количество альтернатив (kol <= 25): ";

    std::cin >> this->kolAlter;
```

```

std::cout << "Вместо пробела используйте символ нижнего подчеркивания \"_\" <<
std::endl;

std::cout << "Введите имя ЦЕЛИ: ";

std::cin >> nameCeli;

for (int i = 1; i <= kolKrit; i++) {

    std::cout << "Введите имя " << i << " критерия: ";

    std::cin >> nameKrit[i-1];

}

char ch = 65;

for (int i = 1; i <= kolAlter; i++) {

    // имя альтернативы:

    nameAlter[i-1] = ch;

    ch++;

}

std::cout << std::endl;


// выделяем память

mainMatrix = new double* [kolKrit];

for (int i = 0; i < kolKrit; i++) {

    mainMatrix[i] = new double [kolAlter];

}

matrixSravneniy = new double* [kolKrit];

for (int i = 0; i < kolKrit; i++) {

    matrixSravneniy[i] = new double [kolKrit];

}

system("pause");

system("cls");

}

void Matrix :: zapolniaem(){

    std::cout << "ЩА будем вводить сравнительные характеристики" << std::endl;

```

```

        std::cout << "для этой цели ->\\"" << nameCeli << "\"\" << std::endl <<
std::endl;

        for(int i = 0; i < kolKrit; i++){

            for(int j = 0; j < kolAlter; j++){

                std::cout << "Введите [" << nameAlter[j] << "]" [" << nameKrit[i] <<
"] - ";

                std::cin >> mainMatrix[i][j];

            }

        }

        std::cout << std::endl;

        system("pause");

        system("cls");


        std::cout << "А теперь будем вводить элементы матрицы попарных" << std::endl;

        std::cout << "сравнений на основе экспертных оценок" << std::endl <<
std::endl;

        for(int i = 0; i < kolKrit; i++){

            for(int j = 0; j < kolKrit; j++){

                std::cout << "Введите [" << nameKrit[i] << "]" [" << j+1 << "]" - ";

                std::cin >> matrixSravneniy[i][j];

            }

        }

        std::cout << std::endl;

        system("pause");

        system("cls");

    }

void Matrix :: reshaem(){

    double arrVectoraPr[kolKrit]; // вектор приоритетов для главной матрицы
(меняется)

    double arrPrioritetov[kolKrit]; // вектор приоритетов для матрицы попарных
сравнений

    std::cout << "Глобальные приоритеты:" << std::endl << std::endl;

```

```

//=====

//сделаем синтез приоритетов из матрицы попарных сравнений

double tempArr[kolKrit]; //временный массив размера кол-ва критериев

for (int i = 0; i<kolKrit; i++) {

    tempArr[i] = 1;

    for(int j = 0; j<kolKrit; j++) {

        tempArr[i] = tempArr[i] * matrixSravneniy[i][j];

    }

    tempArr[i] = pow(tempArr[i], (1.0/kolKrit));

} //вычислили w1, w2, w3, ...

double sum = 0.0;

for (int i = 0; i<kolKrit; i++) {

    sum = sum + tempArr[i];

} //вычислили W

for (int i = 0; i<kolKrit; i++) {

    arrPrioritetov[i] = tempArr[i] / sum;

} //сделали синтез приоритетов

//=====

double matrVectoraPr[kolKrit][kolAlter];

for(int i = 0; i<kolKrit; i++) {

    tempArr[i] = 0.0;

    for(int j = 0; j<kolAlter; j++) {

        double tempChislitel = mainMatrix[i][j];

        matrVectoraPr[i][j] = 1.0;

        for(int j2 = 0; j2<kolAlter; j2++) {

            matrVectoraPr[i][j] = matrVectoraPr[i][j] * (tempChislitel /
mainMatrix[i][j2]);

        }

        matrVectoraPr[i][j] = pow(matrVectoraPr[i][j], 1.0/kolAlter);

```



```

        // std::cout << "вычисленный w[" << i << "]" [" << j << "]" = " <<
matrVectoraPr[i][j] << std::endl; //----->

        tempArr[i] = tempArr[i] + matrVectoraPr[i][j];

    }

}

//=====

for(int kolichествоAlternativ = 0; kolichествоAlternativ < kolAlter;
kolichествоAlternativ++) {

    sum = 0.0;

    for(int i = 0; i < kolKrit; i++){

        arrVectoraPr[i] = (matrVectoraPr[i][kolichествоAlternativ] /
tempArr[i]);

    }

    for (int i = 0; i<kolKrit; i++) {

        sum = sum + (arrVectoraPr[i] * arrPrioritetov[i]);

    } //вычислили глобальный приоритет

    std::cout << "k" << nameAlter[kolichествоAlternativ] << " = " << sum <<
std::endl;

}

std::cout << std::endl << std::endl;

system("pause");

}

Matrix :: ~Matrix() {

    // очистить память выделенную матрицам

    for (int i = 0; i < kolKrit; i++) {

        delete [] mainMatrix[i];

    }

    delete [] mainMatrix;

```

```

for (int i = 0; i < kolKrit; i++) {

    delete [] matrixSravneniy[i];

}

delete [] matrixSravneniy;

}

int main() {

    SetConsoleCP(1251);

    SetConsoleOutputCP(1251);

    system("color B0");

    //создадим обьедок с нашими данными МАИ

    Matrix *MAI = new Matrix();

    MAI->zapolniaem();

    MAI->reshaem();

    delete MAI;

    return 0;

}

```

Результат выполнения программы соответствует ожиданиям и отображен на рисунках 4.7-

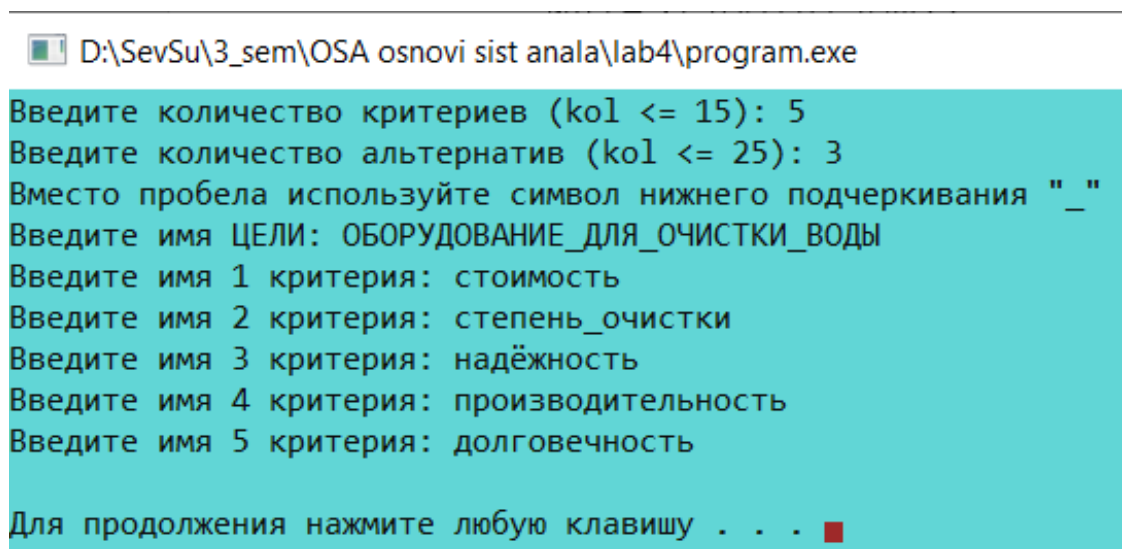


Рисунок 4.7 – Ввод размерности матрицы и имён критериев

```

D:\SevSu\3_sem\OSA osnovi sist anala\lab4\program.exe
ЩА будем вводить сравнительные характеристики
для этой цели ->"ОБОРУДОВАНИЕ_ДЛЯ_ОЧИСТКИ_ВОДЫ"

Введите [A] [стоимость] - 100
Введите [B] [стоимость] - 80
Введите [C] [стоимость] - 50
Введите [A] [степень_очистки] - 98
Введите [B] [степень_очистки] - 95
Введите [C] [степень_очистки] - 90
Введите [A] [надёжность] - 85
Введите [B] [надёжность] - 85
Введите [C] [надёжность] - 70
Введите [A] [производительность] - 0.2
Введите [B] [производительность] - 0.1
Введите [C] [производительность] - 0.05
Введите [A] [долговечность] - 4
Введите [B] [долговечность] - 5
Введите [C] [долговечность] - 5

Для продолжения нажмите любую клавишу . . .

```

Рисунок 4.8 – Заполнение матрицы с критериями альтернатив

```

D:\SevSu\3_sem\OSA osnovi sist anala\lab4\program.exe
А теперь будем вводить элементы матрицы попарных
сравнений на основе экспертных оценок

Введите [стоимость] [1] - 1
Введите [стоимость] [2] - 1
Введите [стоимость] [3] - 0.1666
Введите [стоимость] [4] - 0.3333
Введите [стоимость] [5] - 2
Введите [степень_очистки] [1] - 1
Введите [степень_очистки] [2] - 1
Введите [степень_очистки] [3] - 5
Введите [степень_очистки] [4] - 8
Введите [степень_очистки] [5] - 0.5
Введите [надёжность] [1] - 6
Введите [надёжность] [2] - 0.2
Введите [надёжность] [3] - 1
Введите [надёжность] [4] - 6
Введите [надёжность] [5] - 0.25
Введите [производительность] [1] - 3
Введите [производительность] [2] - 0.125
Введите [производительность] [3] - 0.1666
Введите [производительность] [4] - 1
Введите [производительность] [5] - 3
Введите [долговечность] [1] - 0.5
Введите [долговечность] [2] - 2
Введите [долговечность] [3] - 4
Введите [долговечность] [4] - 0.3333
Введите [долговечность] [5] - 1

Для продолжения нажмите любую клавишу . . .

```

Рисунок 4.9 – Заполнение матрицы попарных сравнений

```

D:\SevSu\3_sem\OSA osnovi sist anala\lab4\program.exe
Глобальные приоритеты:

kA = 0.376636
kB = 0.338592
kC = 0.284772

Для продолжения нажмите любую клавишу . . .

```

Рисунок 4.10 – Результат выполнения программы

Таким образом все программа подтверждает письменные вычисления и доказывает, что вариант “А” действительно является наиболее оптимальным.

Выводы

В ходе выполнения данной лабораторной работы были углублены теоретические знания в области системного анализа, приобретены навыки создания и описания иерархических структур. Изучены такие понятия как цель, критерий и альтернатива. Закреплены навыки разработки программ на языке C++. Полученные во время выполнения лабораторной работы навыки помогут в дальнейшей жизни при необходимости провести системный анализ в какой либо области.