**Министерство образования и науки Российской Федерации Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Севастопольский государственный университет»**

**Лабораторная работа №3**

«МЕТОД РЕШАЮЩИХ МАТРИЦ»

**Вариант 4**

**Выполнил:**

Студент группы ПИ/б-18-1-о

Кисин Игорь Евгеньевич

**Проверил:**

Тлуховская Н. П.

**Севастополь**

**2020**

**3.1 Цель работы**

Углубление теоретических знаний в области системного анализа, исследование способов оценки сложных систем.

Вариант 4

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  |  | |  | 0.2 | 0.3 | 0.5 |  | |  | 0.2 |  | 0.3 | 0.5 | |  | 0.4 | 0.4 |  | 0.2 | |  | 1.0 |  |  |  | |  | 0.1 | 0.1 | 0.2 | 0.5 | | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | |  |  |  |  |  |  | |  | 0.2 |  | 0.4 | 0.4 |  | |  |  | 0.8 |  |  | 0.2 | |  |  |  |  | 0.5 | 0.6 | |  | 0.1 | 0.1 | 0.3 | 0.4 | 0.1 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  |  | |  |  | 0.3 | 0.7 |  | |  | 0.5 | 0.4 |  | 0.1 | |  | 0.2 | 0.2 | 0.2 | 0.4 | |

**3.2 Порядок выполнения**

1. Получить вариант задания - остаток от деления двух последних чисел зачетной книжки на общее количество вариантов задания.

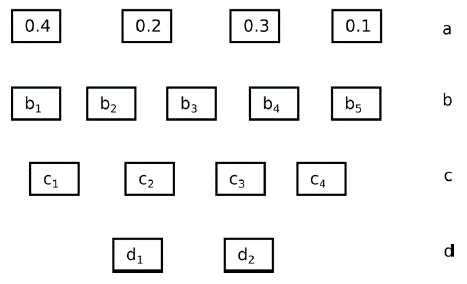
2. На основе исходного варианта задания построить взаимосвязь всех уровней системы.

3. Вычислить влияние вариантов самого нижнего уровня на проектирование всей системы.

4. Написать программу на языке программирования c++, которая решает задачу методом решающих матриц любой размерности.

5. Дополнительное задание: написать программу на языке программирования c++, которая рисует в консоли граф на основе введенных матриц.

**3.3 Ход работы**

Рисунок 1 – Иерархия декомпозиции проблемы

3.3.1 Код программы:

#include <fstream>

#include <iostream>

using namespace std;

class Matrix {

public:

float\*\* M;

int n, m;

Matrix() {

n = 0;

m = 0;

M = nullptr;

}

~Matrix() {

for (int i = 0; i < n; i++) delete[] M[i];

delete[] M;

}

void outputMatrix() {

if(M != nullptr) for (int i = 0; i < n; i++)

{

for (int j = 0; j < m; j++)

cout << M[i][j] << "\t";

cout << "\n";

}

}

void readMatrix(string fileName) {

ifstream in(fileName);

if (in.is\_open())

{

int count = 0;

int temp;

while (!in.eof())

{

in >> temp;

count++;

}

in.seekg(0, ios::beg);

in.clear();

int count\_space = 0;

char symbol;

while (!in.eof())

{

in.get(symbol);

if (symbol == ' ') count\_space++;

if (symbol == '\n') break;

}

in.seekg(0, ios::beg);

in.clear();

n = count / (count\_space + 1);

m = count\_space + 1;

M = new float\* [n];

for (int i = 0; i < n; i++) M[i] = new float[m];

for (int i = 0; i < n; i++)

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

in >> M[i][j];

M[i][j] /= 10;

}

in.close();

}

else

{

cout << "Файл не найден.";

}

}

};

class Mass {

public:

float\* M;

int m;

Mass() {

M = nullptr;

m = 0;

}

~Mass() {

delete[] M;

}

Mass(int \_m) {

M = new float[\_m];

for (int i = 0; i < \_m; i++) {

M[i] = 0;

}

m = \_m;

}

void calculate(Matrix \_M, Mass Prev) {

float M\_SUM = 0;

for (int i = 0; i < \_M.n; i++) {

for (int j = 0; j < Prev.m; j++) {

M[i] += \_M.M[i][j] \* Prev.M[j];

}

M\_SUM += M[i];

}

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

M[i] = M[i] / M\_SUM;

}

}

void output() {

for (int i = 0; i < m; i++)

cout << M[i] << "\t";

cout << "\n";

}

};

int main()

{

setlocale(LC\_ALL, "RUSSIAN");

Mass a(4); //Создание класса весов

//Заполнение весов первого уровня

a.M[0] = 0.4f;

a.M[1] = 0.2f;

a.M[2] = 0.3f;

a.M[3] = 0.1f;

cout << "a: "; a.output(); //Вывод весов уровня А

Matrix B; //Создание класса таблицы

B.readMatrix("B.txt"); //Считывание таблицы из файла

Mass b(B.n); //Создание класса весов второго уровня и инициализация

b.calculate(B, a); //Расчет веса

cout << "b: "; b.output(); //Вывод весов уровня B

Matrix C;

C.readMatrix("C.txt");

Mass c(C.n);

c.calculate(C, b);

cout << "c: "; c.output(); //Вывод весов уровня C

Matrix D;

D.readMatrix("D.txt");

Mass d(D.n);

d.calculate(D, c);

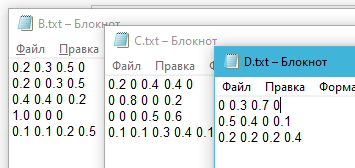
cout << "d: "; d.output(); //Вывод весов уровня D

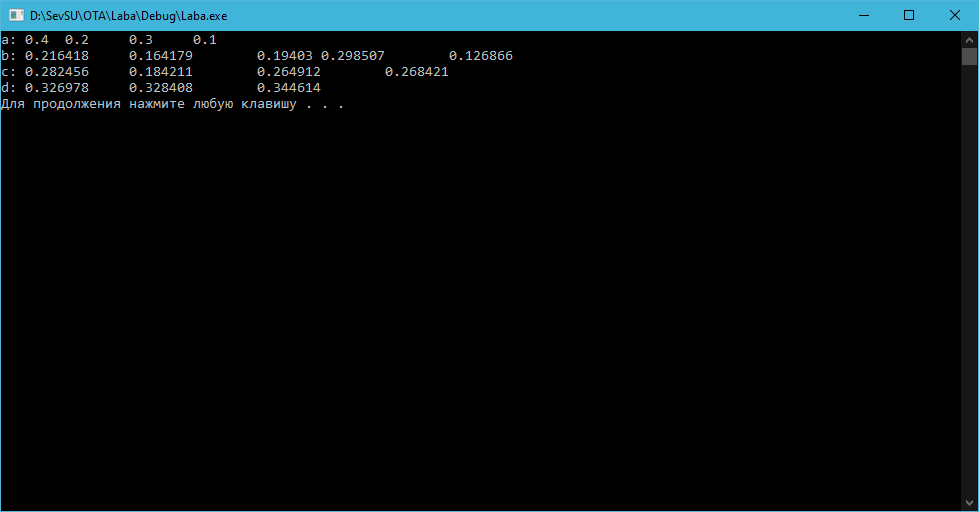
system("pause");

return 0;

}

Условия гибкости программы: Может реализовать расчет задачи методом решающих матриц любой размерности при правильном вводе и инициализации весов первого уровня и соблюдения размерностей таблиц связей. Заполнение таблиц производится через текстовые файлы.

  
Рисунок 2 – Заполнение таблиц перед запуском

  
Рисунок 3 – Выполнение программы

**3.4 Вывод**

В ходе лабораторной работы было оценено влияние факторов нижнего уровня на систему. Было определено, что на уровне B наибольший вес имеет 4 фактор, на уровне C 1, 2 и 4 факторы примерно одинаковы весу, 3 фактор имеет меньший вес. На последнем уровне вес факторов приблизительно равен. Таким образом, был сделан вывод, что D1, D2 и D3 имеют приблизительно равное влияние на проектирование системы.