

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления  
Кафедра Интеллектуальных информационных технологий

ОТЧЁТ  
Лабораторная работа №1

Выполнил:

Кривецкий А.Э.

Проверил:

Никифоров С. А.

Минск 2022

**Тема:** изучение основ объектно-ориентированного программирования на языке программирования C++.

**Задание:** реализовать на языке C++ один из нижеперечисленных вариантов и написать и сгенерировать документацию при помощи doxygen.

**Вариант 1.7.2 :** Вещественная матрица (усложненный).

Описать класс, реализующий тип данных "Вещественная матрица".

file:///D:/PPOIS/FirstLab/doc/html/index.html

**Содержимое файла Matrix.h:**

```
/** \file Matrix.h
 * Объявляется класс
 * Объявляются переменные класса
 * Объявляются методы класса
 */
#pragma once
#include <vector>
#include <iostream>

using namespace std;

class Matrix{
private:
    vector<vector<double>>> b;
    vector<vector<double>>> vec;
public:
    int rows = 0, cols = 0;
    double numb = 0, det = 0;
    vector<vector<double>>> a;

    /// \brief Этот метод реализует создание одной матрицы
    void create_one(Matrix &, int, int);

    /// \brief Этот метод реализует просмотр одной матрицы
    static void show_one(vector<vector<double>>&, int, int);

    /// \brief Этот метод реализует нахождение определителя матрицы
    static void deter(vector<vector<double>>&, int, double);

    /// \brief Этот метод реализует нахождение определителя матрицы
    static double cadet(vector<vector<double>>&, int);

    /// \brief Этот метод реализует нахождение определителя матрицы
    static void new_matr(vector<vector<double>>&, vector<vector<double>>& help, int, int, int);

    /// \brief Этот метод реализует нахождение M-нормы матрицы
    static void m_norm(vector<vector<double>>&, int, int);

    /// \brief Этот метод реализует нахождение L-нормы матрицы
    static void l_norm(vector<vector<double>>&, int, int);

    /// \brief Этот метод реализует нахождение K-нормы матрицы
    static void k_norm(vector<vector<double>>&, int, int);
```

```

Matrix& operator + (Matrix& right) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] = a[i][j] + right.a[i][j];
        }
    }
    return *(this);
}

Matrix& operator += (Matrix& right) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] += right.a[i][j];
        }
    }
    return *(this);
}

Matrix& operator - (Matrix& right) {
    //Matrix tmp = *this;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] = a[i][j] - right.a[i][j];
        }
    }
    return *(this);
}

Matrix& operator -= (Matrix& right) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] -= right.a[i][j];
        }
    }
    return *(this);
}

```

```

Matrix& operator + (double& right) {
    //Matrix tmp = *this;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] = a[i][j] + right;
        }
    }
    return *(this);
}

Matrix& operator += (double right) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] += right;
        }
    }
    return *(this);
}

Matrix& operator - (double& right) {
    //Matrix tmp = *this;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] = a[i][j] - right;
        }
    }
    return *(this);
}

```

```

Matrix& operator -= (double right) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] -= right;
        }
    }
    return *(this);
}

Matrix& operator / (double& right) {
    //Matrix tmp = *this;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] = a[i][j] / right;
        }
    }
    return *(this);
}

Matrix& operator /= (double right) {
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] /= right;
        }
    }
    return *(this);
}

```

```

Matrix& operator * (double& right) {
    Matrix tmp = *this;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] = a[i][j] * right;
        }
    }
    return *(this);
}

Matrix& operator *= (double& right) {
    Matrix tmp = *this;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            this->a[i][j] *= right;
        }
    }
    return *(this);
}

```

```

Matrix& operator * (Matrix& right) {
    b.resize(_Newsize: rows);
    for (int i = 0; i < rows; i++) {
        b[i].resize(_Newsize: cols);
    }
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            for (int x = 0; x < cols; x++) {
                b[i][j] += a[i][x] * right.a[x][j];
            }
        }
    }
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            a[i][j] = b[i][j];
        }
    }
    return *this;
}

```

```

Matrix& operator ^ (double right) {
    b.resize(_Nrows; rows);
    for (int i = 0; i < rows; i++) {
        b[i].resize(_Ncols; cols);
    }

    vec.resize(_Nrows; rows);
    for (int i = 0; i < rows; i++) {
        vec[i].resize(_Ncols; cols);
    }

    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            vec[i][j] = a[i][j];
        }
    }

    for (int k = 0; k < right - 1; k++)
    {
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                for (int x = 0; x < cols; x++) {
                    b[i][j] += vec[i][x] * a[x][j];
                }
            }
        }
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                a[i][j] = b[i][j];
                b[i][j] = 0;
            }
        }
    }
    return *this;
}

```

```

Matrix& operator ^=(double right) {
    b.resize(_Newsize, rows);
    for (int i = 0; i < rows; i++) {
        b[i].resize(_Newsize, cols);
    }

    vec.resize(_Newsize, rows);
    for (int i = 0; i < rows; i++) {
        vec[i].resize(_Newsize, cols);
    }

    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            vec[i][j] = a[i][j];
        }
    }

    for (int k = 0; k < right - 1; k++)
    {
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
            {
                for (int x = 0; x < cols; x++) {
                    b[i][j] += vec[i][x] * a[x][j];
                }
            }
        }
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                a[i][j] = b[i][j];
                b[i][j] = 0;
            }
        }
    }
    return *this;
}

```

```

Matrix& operator ++ (int) {
    this->a.resize(_Newsize: this->rows + 1);
    for (int i = 0; i < this->rows + 1; i++) {
        this->a[i].resize(_Newsize: this->cols + 1);
    }
    for (int i = 0; i < this->rows + 1; i++) {
        for (int j = 0; j < this->cols + 1; j++) {
            double d;
            cin >> d;
            this->a[i][j] = d;
        }
    }
    return *this;
}

Matrix& operator -- (int) {
    this->a.resize(_Newsize: this->rows - 1);
    for (int i = 0; i < this->rows - 1; i++) {
        this->a[i].resize(_Newsize: cols - 1);
    }
    for (int i = 0; i < this->rows - 1; i++) {
        for (int j = 0; j < this->cols - 1; j++) {
            double d;
            cin >> d;
            this->a[i][j] = d;
        }
    }
    return *this;
}

void ShowMatrix()
{
    for (int i = 0; i < a.size(); i++)
    {
        for (int j = 0; j < a[i].size(); j++)
        {
            cout << a[i][j] << '\t';
        }
        cout << '\n';
    }
}
};

```

В файле Matrix.h происходит объявление класса Matrix, перечисление его полей и методов.

**Содержимое файла Matrix.cpp:**



```

/** \file Matrix.cpp
 * Реализуются методы класса
 * */
#include "Matrix.h"
#include <iostream>
#include <vector>

using namespace std;

/// \code
void Matrix::create_one(Matrix &mat, int row, int col) {
    mat.a.clear();
    mat.a.resize(_Nsize:row);
    for (int i = 0; i < row; i++) {
        mat.a[i].resize(_Nsize:col);
    }

    vector<double> help;
    cout << "Введите элементы матрицы:" << endl;
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            double c;
            cin >> c;
            help.push_back(_Val:c);
            mat.a[i][j] = help[0];
            help.clear();
        }
    }
}

}

/// \code
void Matrix::show_one(vector<vector<double>>& a, int row, int col) {

    cout << "Вывод матрицы:" << endl;
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            cout << a[i][j] << "\t";
        }
        cout << endl;
    }
}

}

```

```

void Matrix::new_matr(vector<vector<double>>& a, vector<vector<double>>& help, int i, int j, int row) {
    int k1, k2, d1, d2;
    d1 = 0;
    for (k1 = 0; k1 < row - 1; k1++) {
        if (k1 == i) {
            d1 = 1;
        }
        d2 = 0;
        for (k2 = 0; k2 < row - 1; k2++) {
            if (k2 == j) {
                d2 = 1;
            }
            help[k1][k2] = a[k1 + d1][k2 + d2];
        }
    }
}

/// \code
double Matrix::cadet(vector<vector<double>>& a, int row) {
    int j, k, n;
    double det;
    vector<vector<double>> help;
    help.resize(_Nsize:row);
    for (int i = 0; i < row; i++)
        help[i].resize(_Nsize:row);
    j = 0; det = 0;
    k = 1;
    n = row - 1;
    if (row < 1) cout << "Определитель вычислить невозможно!";
    if (row == 1) {
        det = a[0][0];
        return det;
    }
    if (row == 2) {
        det = a[0][0] * a[1][1] - (a[1][0] * a[0][1]);
        return det;
    }
    if (row > 2) {
        for (int i = 0; i < row; i++) {
            Matrix::new_matr(a, help, i, j=0, row);
            cout << a[i][j] << endl;
            Matrix::show_one(a, help, row:n, col:n);
            det = det + k * a[i][0] * cadet(a, help, row:n);
            k = -k;
        }
    }
    return det;
}

```

```

/// \code
void Matrix::deter(vector<vector<double>>& a, int row, double det) {
    det = Matrix::cadet(a, row);
    cout << "Определитель матрицы равен det = " << det << endl;
}

/// \code
void Matrix::m_norm(vector<vector<double>>& a, int row, int col) {
    vector<double> res;
    res.resize(res.size()+row);
    double cou = 0;

    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            cou += a[i][j] ;
        }
        res[i] = cou;
        cou = 0;
    }

    double max = res[0];
    for (int i = 0; i < row; i++)
    {
        if (max < res[i]) {
            max = res[i];
        }
    }

    cout << "M-норма матрицы:" << endl;
    cout << max << endl;
    show_one(a, row, col);
}

```

```

/// \code
void Matrix::l_norm(vector<vector<double>>& a, int row, int col) {

    vector<double> res;
    res.resize(_Nrows; row);
    double cou = 0;

    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            cou += a[j][i];
        }
        res[i] = cou;
        cou = 0;
    }

    double max = res[0];
    for (int i = 0; i < row; i++)
    {
        if (max < res[i]) {
            max = res[i];
        }
    }

    cout << "L-норма матрицы:" << endl;
    cout << max << endl;
    show_one(a, row, col);
}

/// \code
void Matrix::k_norm(vector<vector<double>>& a, int row, int col) {

    double cou = 0;

    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            cou += pow(a[i][_Ncols[j]], _Ncols[j]);
        }
    }

    cou = sqrt(cou);

    cout << "K-норма матрицы:" << endl;
    cout << cou << endl;
    show_one(a, row, col);
}

```

В файле Matrix.cpp содержится реализация методов класса Matrix.

**Содержимое файла laba1.cpp:**

[illegible]

```

        println("%i%i^2")
        if (mail.ram == mail.ram || mail.sfx == mail.sfx) {
            result = mail + mail2
            result.showFields()
            result.x.clear()
        }
        else {
            mail += "Написано письмо с адресом страницы 4 и 8 первая буква опечатана !!!" + mail;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail.ram;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail.sfx;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail2.ram;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail2.sfx;
            while (n) {
                mail += "2 - страница 4.7" + mail + "2 - страница 4.7" + mail + "2 - страница" + mail;
                sfx += n;
                switch (n) {
                    case 1:
                        mail += "Написано письмо с адресом страницы 4.7" + mail;
                        sfx += mail.ram;
                        mail += "Написано письмо с адресом страницы 4.7" + mail;
                        sfx += mail.sfx;
                        mail.remove_pos(mail, mail.ram, mail.sfx);
                        break;
                    case 2:
                        mail += "Написано письмо с адресом страницы 4.7" + mail;
                        sfx += mail2.ram;
                        mail += "Написано письмо с адресом страницы 4.7" + mail;
                        sfx += mail2.sfx;
                        mail2.remove_pos(mail2, mail2.ram, mail2.sfx);
                        break;
                    case 3:
                        n = false;
                        println("%i%i^2");
                        break;
                }
            }
        }
        break;
    }
}

// case 5:
println("%i%i^2")
mail += "2 - первая буква с адресом 4.7" + mail + "2 - первая буква с адресом 4.7" + mail;
sfx += n;
switch (n) {
    case 1:
        mail += "Написано письмо" + mail;
        sfx += ram;
        result = mail + ram;
        result.showFields()
        result.x.clear()
        break;
    case 2:
        mail += "Написано письмо" + mail;
        sfx += ram;
        result = mail + ram;
        result.showFields()
        result.x.clear()
        break;
}
break;

// case 6:
println("%i%i^2")
mail += "2 - 4-й" + mail + "2 - 4-й" + mail;
sfx += n;
switch (n) {
    case 1:
        if (mail.ram == mail.ram || mail.sfx == mail.sfx) {
            result = mail + mail2
            result.showFields()
            result.x.clear()
        }
        else {
            println("%i%i^2")
            mail += "Написано письмо с адресом страницы 4 и 8 первая буква опечатана !!!" + mail;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail.ram;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail2.sfx;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail2.ram;
            mail += "Написано письмо с адресом страницы 4.7" + mail;
            sfx += mail2.sfx;
        }
    }
}

```

```

1210   var m1 = "Заголовок сообщения и путь к файлу 101" vs m2;
1211   s1 := m11.com;
1212   m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1213   s1 := m12.com;
1214   m11.send_m(m1, m11.com, m11.s1);
1215   m12.send_m(m1, m12.com, m11.s1);
1216 }
1217 break;
1218
1219 case 2:
1220   if (m11.com = m12.com && m11.s1 = m12.s1) {
1221     result := m12 - m11;
1222     result.ShowMessage();
1223     result := Close();
1224   }
1225   else {
1226     options("101");
1227     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1228     m11 := "Заголовок сообщения и путь к файлу 102" vs m2;
1229     s1 := m11.com;
1230     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1231     s1 := m12.com;
1232     m11.send_m(m1, m11.com, m11.s1);
1233     m12.send_m(m1, m12.com, m11.s1);
1234   }
1235   break;
1236 }
1237 break;
1238
1239 case 3:
1240   options("102");
1241   m1 := "1" - параметр меню и путь к файлу 101" vs m2 - параметр меню и путь к файлу 102";
1242   s1 := m1.com;
1243   m11k := 0;
1244
1245   case 1:
1246     m1 := "Заголовок меню" vs m2;
1247     s1 := m1.com;
1248     result := m11 - m1;
1249     result.ShowMessage();
1250     result := Close();
1251     break;
1252
1253   case 2:
1254     m1 := "Заголовок меню" vs m2;
1255     s1 := m1.com;
1256     result := m12 - m1;
1257     result.ShowMessage();
1258     result := Close();
1259     break;
1260 }
1261 break;
1262
1263 case 4:
1264   options("103");
1265   m1 := "1" - m2; vs m2 := "2" - m2; vs m2;
1266   s1 := m1.com;
1267   m11k := 0;
1268
1269   case 1:
1270     if (m11.s1 = m12.com) {
1271       result := m11 - m1;
1272       result.ShowMessage();
1273       result := Close();
1274     }
1275     else {
1276       m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1277       s1 := m11.com;
1278       m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1279       s1 := m12.s1;
1280       m11.send_m(m1, m11.com, m11.s1);
1281       m12.send_m(m1, m12.com, m11.s1);
1282     }
1283     break;
1284
1285   case 2:
1286     if (m12.s1 = m11.com) {
1287       result := m12 - m1;
1288       result.ShowMessage();
1289       result := Close();
1290     }
1291     else {
1292       m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1293       s1 := m11.com;
1294       m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1295       s1 := m12.s1;
1296       m11.send_m(m1, m11.com, m11.s1);
1297       m12.send_m(m1, m12.com, m11.s1);
1298     }
1299     break;
1300 }
1301 break;
1302
1303 case 5:
1304   if (m12.s1 = m11.com) {
1305     result := m12 - m1;
1306     result.ShowMessage();
1307     result := Close();
1308   }
1309   else {
1310     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1311     s1 := m11.com;
1312     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1313     s1 := m12.s1;
1314     m11.send_m(m1, m11.com, m11.s1);
1315     m12.send_m(m1, m12.com, m11.s1);
1316   }
1317   break;
1318 }
1319 break;
1320
1321 case 6:
1322   if (m12.s1 = m11.com) {
1323     result := m12 - m1;
1324     result.ShowMessage();
1325     result := Close();
1326   }
1327   else {
1328     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1329     s1 := m11.com;
1330     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1331     s1 := m12.s1;
1332     m11.send_m(m1, m11.com, m11.s1);
1333     m12.send_m(m1, m12.com, m11.s1);
1334   }
1335   break;
1336 }
1337 break;
1338
1339 case 7:
1340   if (m12.s1 = m11.com) {
1341     result := m12 - m1;
1342     result.ShowMessage();
1343     result := Close();
1344   }
1345   else {
1346     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1347     s1 := m11.com;
1348     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1349     s1 := m12.s1;
1350     m11.send_m(m1, m11.com, m11.s1);
1351     m12.send_m(m1, m12.com, m11.s1);
1352   }
1353   break;
1354 }
1355 break;
1356
1357 case 8:
1358   if (m12.s1 = m11.com) {
1359     result := m12 - m1;
1360     result.ShowMessage();
1361     result := Close();
1362   }
1363   else {
1364     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1365     s1 := m11.com;
1366     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1367     s1 := m12.s1;
1368     m11.send_m(m1, m11.com, m11.s1);
1369     m12.send_m(m1, m12.com, m11.s1);
1370   }
1371   break;
1372 }
1373 break;
1374
1375 case 9:
1376   if (m12.s1 = m11.com) {
1377     result := m12 - m1;
1378     result.ShowMessage();
1379     result := Close();
1380   }
1381   else {
1382     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1383     s1 := m11.com;
1384     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1385     s1 := m12.s1;
1386     m11.send_m(m1, m11.com, m11.s1);
1387     m12.send_m(m1, m12.com, m11.s1);
1388   }
1389   break;
1390 }
1391 break;
1392
1393 case 10:
1394   if (m12.s1 = m11.com) {
1395     result := m12 - m1;
1396     result.ShowMessage();
1397     result := Close();
1398   }
1399   else {
1400     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1401     s1 := m11.com;
1402     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1403     s1 := m12.s1;
1404     m11.send_m(m1, m11.com, m11.s1);
1405     m12.send_m(m1, m12.com, m11.s1);
1406   }
1407   break;
1408 }
1409 break;
1410
1411 case 11:
1412   if (m12.s1 = m11.com) {
1413     result := m12 - m1;
1414     result.ShowMessage();
1415     result := Close();
1416   }
1417   else {
1418     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1419     s1 := m11.com;
1420     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1421     s1 := m12.s1;
1422     m11.send_m(m1, m11.com, m11.s1);
1423     m12.send_m(m1, m12.com, m11.s1);
1424   }
1425   break;
1426 }
1427 break;
1428
1429 case 12:
1430   if (m12.s1 = m11.com) {
1431     result := m12 - m1;
1432     result.ShowMessage();
1433     result := Close();
1434   }
1435   else {
1436     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1437     s1 := m11.com;
1438     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1439     s1 := m12.s1;
1440     m11.send_m(m1, m11.com, m11.s1);
1441     m12.send_m(m1, m12.com, m11.s1);
1442   }
1443   break;
1444 }
1445 break;
1446
1447 case 13:
1448   if (m12.s1 = m11.com) {
1449     result := m12 - m1;
1450     result.ShowMessage();
1451     result := Close();
1452   }
1453   else {
1454     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1455     s1 := m11.com;
1456     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1457     s1 := m12.s1;
1458     m11.send_m(m1, m11.com, m11.s1);
1459     m12.send_m(m1, m12.com, m11.s1);
1460   }
1461   break;
1462 }
1463 break;
1464
1465 case 14:
1466   if (m12.s1 = m11.com) {
1467     result := m12 - m1;
1468     result.ShowMessage();
1469     result := Close();
1470   }
1471   else {
1472     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1473     s1 := m11.com;
1474     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1475     s1 := m12.s1;
1476     m11.send_m(m1, m11.com, m11.s1);
1477     m12.send_m(m1, m12.com, m11.s1);
1478   }
1479   break;
1480 }
1481 break;
1482
1483 case 15:
1484   if (m12.s1 = m11.com) {
1485     result := m12 - m1;
1486     result.ShowMessage();
1487     result := Close();
1488   }
1489   else {
1490     m1 := "Заголовок сообщения и путь к файлу 101" vs m2;
1491     s1 := m11.com;
1492     m1 := "Заголовок сообщения и путь к файлу 102" vs m2;
1493     s1 := m12.s1;
1494     m11.send_m(m1, m11.com, m11.s1);
1495     m12.send_m(m1, m12.com, m11.s1);
1496   }
1497   break;
1498 }
1499 break;
1500
1501 case 16:
1502   if (m12.s1 = m11.com) {
1503     result := m12 - m1;
1504     result.ShowMessage();
150
```





```

        else if (degree)
        {
            result = null;
            result.ShuffleList();
            result.Clear();
        }
        else {
            result = "Because you attempted to a system gamma beta suggested 1117 vs null;
            result = "Because suggested system suggested 1117 vs null;
            else if null.result;
            result = "Because suggested system suggested 1117 vs null;
            else if null.result;
            result.ShuffleList();
        }
        break;
    }
}
break;

case 11:
    system("cls");
    result = "1 = suggested system 1117 vs null vs 12 = suggested system 1117 vs null;
    else if null;
    switch (x) {

        case 1:
            if (null.result == null.result) {
                result.Clear();
            }
            else {
                result = "Because you attempted to a system gamma beta suggested 1117 vs null;
                result = "Because suggested system suggested 1117 vs null;
                else if null.result;
                result = "Because suggested system suggested 1117 vs null;
                else if null.result;
                result.ShuffleList();
            }
            break;

        case 2:
            if (null.result == null.result) {
                result.Clear();
            }
            else {
                result = "Because you attempted to a system gamma beta suggested 1117 vs null;
                result = "Because suggested system suggested 1117 vs null;
                else if null.result;
                result = "Because suggested system suggested 1117 vs null;
                else if null.result;
                result.ShuffleList();
            }
            break;
    }
}
break;

case 12:
    system("cls");
    result = "1 = system suggested 1117 vs null vs 12 = system suggested 1117 vs null;
    else if null;
    switch (x) {

        case 1:
            result = "1 = system suggested 1117 vs null vs 12 = 1-system suggested 1117 vs null vs 12 = 0-system suggested 1117 vs null;
            else if null;
            switch (x) {

                case 1:
                    result.Clear();
                    break;

                case 2:
                    result.Clear();
                    break;

                case 3:
                    result.Clear();
                    break;
            }
            break;

        case 2:
            result = "1 = system suggested 1117 vs null vs 12 = 1-system suggested 1117 vs null vs 12 = 0-system suggested 1117 vs null;
            else if null;
            switch (x) {

                case 1:
                    result.Clear();
                    break;

                case 2:
                    result.Clear();
                    break;
            }
            break;
    }
}
break;

```

```

        state 2:
            Mailbox1_send(mail2_a, mail2_recv, mail2_eof);
            break;

        state 3:
            Mailbox1_send(mail2_a, mail2_recv, mail2_eof);
            break;
    }
    break;
}

state 10:
    sprintf("%i\n",
    k = k + 1;
    break;
}

break;

state 2:
while (k == true) {
    for m;
    send vs "10 = process message" vs mail vs "12 = opening message" vs mail vs "13 = open message & message 0" vs mail vs "14 = open message & message" vs mail vs "15 = process message" vs mail vs "16 = process message & message" vs mail;
    send vs "17 = process message & message 0" vs mail vs "18 = process message & message" vs mail vs "19 = process message & message" vs mail vs "20 = process message & message" vs mail;
    send vs "21 = process message & message" vs mail vs "22 = open message" vs mail vs "23 = message" vs mail;
    side vs m;
    switch (m) {
        state 1:
            sprintf("%i\n",
            while (k) {
                send vs "10 = process message" vs mail vs "12 = message 0" vs mail vs "13 = message" vs mail;
                side vs m;
                switch (m) {
                    state 1:
                        send vs "10 = process message" vs mail;
                        side vs mail_recv;
                        send vs "10 = process message" vs mail;
                        side vs mail_eof;
                        mail_send_mail(mail_recv, mail_eof);
                        break;

                    state 2:
                        send vs "10 = process message" vs mail;
                        side vs mail_recv;
                        send vs "10 = process message" vs mail;
                        side vs mail_eof;
                        mail_send_mail(mail_recv, mail_eof);
                        break;

                    state 3:
                        sprintf("%i\n",
                        k = k + 1;
                        break;
                    }
                }
            }
            break;

        state 2:
            sprintf("%i\n",
            while (k) {
                send vs "10 = message 0" vs mail vs "12 = message 0" vs mail vs "13 = message" vs mail;
                side vs m;
                switch (m) {
                    state 1:
                        mail_send_mail();
                        break;

                    state 2:
                        mail_send_mail();
                        break;

                    state 3:
                        sprintf("%i\n",
                        k = k + 1;
                        break;
                    }
                }
            }
            break;
    }
}

```

```

    }
    break;

case 3:

    system("cls");
    if (mat1.sum == mat2.sum || mat1.sfa == mat2.sfa) {
        mat1 = mat2;
        mat1.showMatrix();
    }
    else {
        cout << "Hesablanan xipos a avrodlan narqam 3 a 8 girana bura aqarmanan !!!" << endl;
        cout << "Hesaba namozotidan xipos narqam 3.7" << endl;
        cin >> mat1.sum;
        cout << "Hesaba namozotidan avrodlan narqam 3.7" << endl;
        cin >> mat1.sfa;
        cout << "Hesaba namozotidan xipos narqam 3.7" << endl;
        cin >> mat2.sum;
        cout << "Hesaba namozotidan avrodlan narqam 3.7" << endl;
        cin >> mat2.sfa;
        while (n) {
            cout << "3 - narqam 3.7" << endl << "2 - narqam 3.7" << endl << "2 - narqam 3.7" << endl;
            cin >> n;
            switch (n) {
                case 1:
                    cout << "Hesaba namozotidan xipos narqam 3.7" << endl;
                    cin >> mat1.sum;
                    cout << "Hesaba namozotidan avrodlan narqam 3.7" << endl;
                    cin >> mat1.sfa;
                    mat1.showMatrix();
                    break;

                case 2:
                    cout << "Hesaba namozotidan xipos narqam 3.7" << endl;
                    cin >> mat2.sum;
                    cout << "Hesaba namozotidan avrodlan narqam 3.7" << endl;
                    cin >> mat2.sfa;
                    mat2.showMatrix();
                    break;

                case 3:
                    M = false;
                    system("cls");
                    break;
            }
        }
    }
    break;

case 4:

    system("cls");
    cout << "1 - xipos namoz a narqam 3.7" << endl << "2 - xipos namoz a narqam 3.7" << endl;
    cin >> n;
    switch (n) {
        case 1:
            cout << "Hesaba namoz" << endl;
            cin >> sum;
            mat1 = sum;
            mat1.showMatrix();
            break;

        case 2:
            cout << "Hesaba namoz" << endl;
            cin >> sum;
            mat2 = sum;
            mat2.showMatrix();
            break;
    }
    break;

case 5:

    system("cls");
    cout << "1 - 3*3" << endl << "2 - 3*3" << endl;
    cin >> n;
    switch (n) {
        case 1:
            if (mat1.sum == mat2.sum || mat1.sfa == mat2.sfa) {
                mat1 = mat2;
                result.showMatrix();
                result.a.clear();
            }
            else {
                system("cls");
                cout << "Hesablanan xipos a avrodlan narqam 3 a 8 girana bura aqarmanan !!!" << endl;
                cout << "Hesaba namozotidan xipos narqam 3.7" << endl;
                cin >> mat1.sum;
                cout << "Hesaba namozotidan avrodlan narqam 3.7" << endl;
                cin >> mat1.sfa;
                cout << "Hesaba namozotidan xipos narqam 3.7" << endl;
            }
    }
}

```

```

    end;
    if (not isNull(
        mail -- "Введите номерную запись маршрута 8-9" vs mail;
        sda vs mail.rom;
        mail vs "Введите номерную запись маршрута 8-9" vs mail;
        sda vs mail.rom;
        mail -- mail.roma;
        mail.roma_ano(mail, mail.rom, mail.roma);
        mail.roma_ano(mail, mail.rom, mail.roma);
    )
    )
    break;
}

note 2:
if (mail.rom -- mail.rom || mail.roma -- mail.roma) {
    mail -- mail;
    mail.ShiftLeft();
}
else {
    system("%c");
    mail vs "Введите номер и площадь маршрута 8 и 9 по мере необходимости 111" vs mail;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.rom;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.roma;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.rom;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.roma;
    mail.roma_ano(mail, mail.rom, mail.roma);
    mail.roma_ano(mail, mail.rom, mail.roma);
}
break;
}
}
break;
}

note 3:
system("%c");
mail vs "1 - площадь маршрута 8 маршрута 8-9" vs mail vs "2 - площадь маршрута 8-9" vs mail;
sda vs s;
mail(s) {
}

note 1:
mail vs "Введите номер" vs mail;
sda vs rom;
mail -- rom;
mail.ShiftLeft();
break;
}

note 2:
mail vs "Введите номер" vs mail;
sda vs rom;
mail -- rom;
mail.ShiftLeft();
break;
}
}
break;
}

note 7:
system("%c");
mail vs "1 - 8-9" vs mail vs "2 - 8-9" vs mail;
sda vs s;
mail(s) {
}

note 1:
if (mail.roma vs mail.rom) {
    result = mail + mail;
    result.ShiftLeft();
    result = result;
}
else {
    mail vs "Введите площадь маршрута 8 площадь маршрута 8 и площадь маршрута 8 111" vs mail;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.rom;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.roma;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.rom;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.roma;
    mail.roma_ano(mail, mail.rom, mail.roma);
    mail.roma_ano(mail, mail.rom, mail.roma);
}
break;
}

note 2:
if (mail.roma vs mail.rom) {
    result = mail + mail;
    result.ShiftLeft();
    result = result;
}
else {
    mail vs "Введите площадь маршрута 8 площадь маршрута 8 и площадь маршрута 8 111" vs mail;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.rom;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.roma;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.rom;
    mail vs "Введите номерную запись маршрута 8-9" vs mail;
    sda vs mail.roma;
    mail.roma_ano(mail, mail.rom, mail.roma);
    mail.roma_ano(mail, mail.rom, mail.roma);
}
break;
}
}

```

```

    end, vs "Integer overflow upon stepsize h." vs end;
    size := null.run;
    end, vs "Integer overflow on condition stepsize h." vs end;
    size := null.size;
    null.verify_err(null, null.run, null.size);
    null.verify_err(null, null.run, null.size);
  }
  break;
}
break;

case 8:
  println("%i%", i);
  end, vs "2 - stepsize must be stepsize h." vs end, vs "2 - stepsize must be stepsize h." vs end;
  size := 0;
  while (k) {
    case 1:
      end, vs "Integer must" vs end;
      size := size;
      null := size;
      null.verify_err();
      break;
    case 2:
      end, vs "Integer must" vs end;
      size := size;
      null := size;
      null.verify_err();
      break;
    }
  }
  break;

case 9:
  println("%i%", i);
  end, vs "2 - integer stepsize h is wrong." vs end, vs "2 - integer stepsize h is wrong." vs end;
  size := 0;
  while (k) {
    case 1:
      end, vs "Integer must" vs end;
      size := size;
      null := size;
      null.verify_err();
      break;
    case 2:
      end, vs "Integer must" vs end;
      size := size;
      null := size;
      null.verify_err();
      break;
    }
  }
  break;

case 10:
  println("%i%", i);
  end, vs "2 - integer stepsize h is wrong." vs end, vs "2 - integer stepsize h is wrong." vs end;
  size := 0;
  while (k) {
    case 1:
      if (null.run == null.size) {
        end, vs "Integer overflow." vs end;
        size := degree;
        null := degree;
        null.verify_err();
      }
      else {
        end, vs "Integer overflow of a wrong guess from stepsize h." vs end;
        end, vs "Integer overflow upon stepsize h." vs end;
        size := null.run;
        end, vs "Integer overflow on condition stepsize h." vs end;
        size := null.size;
        null.verify_err(null, null.run, null.size);
        null.verify_err(null, null.run, null.size);
      }
      break;
    case 2:
      if (null.run == null.size) {
        end, vs "Integer overflow." vs end;
        size := degree;
        null := degree;
        null.verify_err();
      }
      else {
        end, vs "Integer overflow of a wrong guess from stepsize h." vs end;
        end, vs "Integer overflow upon stepsize h." vs end;
        size := null.run;
        end, vs "Integer overflow on condition stepsize h." vs end;

```

```

        side <= mail.side;
        mail.create_new(mail2, mail2.room, mail2.side);
    }
    break;
}
break;

state 11:
update("%i\n");
mail <= "1 - supersonic supersonic 5.7" <= mail <= "2 - supersonic supersonic 5.7" <= mail;
side <= m;
update("%i\n");

state 1:
if (mail2.room <= mail2.side) {
    Mail::delete(mail2.a, mail2.room, mail2.side);
}
else {
    mail <= "Supersonic supersonic of a supersonic gun has happened (11)" <= mail;
    mail <= "Supersonic supersonic supersonic 5.7" <= mail;
    side <= mail2.room;
    mail <= "Supersonic supersonic supersonic 5.7" <= mail;
    side <= mail2.side;
    mail.create_new(mail2, mail2.room, mail2.side);
}
break;

state 2:
if (mail2.room <= mail2.side) {
    Mail::delete(mail2.a, mail2.room, mail2.side);
}
else {
    mail <= "Supersonic supersonic of a supersonic gun has happened (11)" <= mail;
    mail <= "Supersonic supersonic supersonic 5.7" <= mail;
    side <= mail2.room;
    mail <= "Supersonic supersonic supersonic 5.7" <= mail;
    side <= mail2.side;
    mail.create_new(mail2, mail2.room, mail2.side);
}
break;
}
break;

state 12:
update("%i\n");
mail <= "1 - supersonic supersonic 5.7" <= mail <= "2 - supersonic supersonic 5.7" <= mail;
side <= m;
update("%i\n");

state 1:
mail <= "1 - supersonic supersonic 5.7" <= mail <= "2 - supersonic supersonic 5.7" <= mail <= "3 - supersonic supersonic 5.7" <= mail;
side <= m;
update("%i\n");

state 1:
    Mail::create_new(mail2.a, mail2.room, mail2.side);
    break;

state 2:
    Mail::create_new(mail2.a, mail2.room, mail2.side);
    break;

state 3:
    Mail::create_new(mail2.a, mail2.room, mail2.side);
    break;
}
break;

state 2:
mail <= "1 - supersonic supersonic 5.7" <= mail <= "2 - supersonic supersonic 5.7" <= mail <= "3 - supersonic supersonic 5.7" <= mail;
side <= m;
update("%i\n");

state 1:
    Mail::create_new(mail2.a, mail2.room, mail2.side);
    break;

state 2:
    Mail::create_new(mail2.a, mail2.room, mail2.side);
    break;

state 3:
    Mail::create_new(mail2.a, mail2.room, mail2.side);
    break;
}
break;
}
break;

```

```

    }
    break;

state 11:
    update("%i\n");
    k = false;
    break;
}
break;

state 1:
mail <= "1 - 5.7" <= mail <= "2 - 5.7" <= mail <= "3 - 5.7" <= mail <= "4 - 5.7" <= mail;
side <= m;
update("%i\n");

state 1:
    mail1.threatLevel();
    mail1++;
    mail1.threatLevel();
    break;

state 2:
    mail1.threatLevel();
    mail1++;
    mail1.threatLevel();
    break;

state 3:
    mail2.threatLevel();
    mail2++;
    mail2.threatLevel();
    break;

state 4:
    mail3.threatLevel();
    mail3++;
    mail3.threatLevel();
    break;
}
break;

state 5:
k = false;
update("%i\n");
break;
}
}
}

```

Файл FirstLab.cpp содержит основную функцию main, реализующая ход выполнения программы.

### Стартовое меню в консоли:

```
Выберете знаки:  
1 - +, -, *, /, ^ :  
2 - +=, -=, *=, /=, ^= :  
3 - ++ или -- :  
4 - выход:
```

```
1 - редактирование матрицы:  
2 - просмотр матрицы:  
3 - сумма матрицы A и матрицы B:  
4 - сумма матрицы с числом:  
5 - разность двух матриц:  
6 - разность матрицы с числом:  
7 - произведение матрицы A и матрицы B:  
8 - произведение матрицы с числом:  
9 - деление матрицы на число:  
10 - возведение матрицы в степень:  
11 - детерминант(определитель) матрицы:  
12 - норма матрицы:  
13 - выход:
```

```
1 - A++  
2 - A--  
3 - B++  
4 - B--
```

### Завершение программы:

```
D:\PPOIS\FirstLab\x64\Debug\FirstLab.exe (процесс 20996) завершил работу с кодом 0.  
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" -> "Параметры" -> "Отладка" -> "Автоматически закрыть консоль при остановке отладки".  
Нажмите любую клавишу, чтобы закрыть это окно: _
```

### Вывод:

Я изучил основы объектно-ориентированного программирования на языке программирования C++, а также реализовал на языке C++ класс вещественных матриц и написать и сгенерировать документацию при помощи doxygen.

