**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ**

**«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»**

Институт компьютерных наук и технологического образования

Кафедра компьютерных технологий и электронного обучения

**КУРСОВАЯ РАБОТА**

ТРЕНАЖЕР ПО ЧИСЛЕННЫМ МЕТОДАМ РЕШЕНИЯ ЗАДАЧ ЛИНЕЙНОЙ АЛГЕБРЫ

Направление подготовки: «Информатика и вычислительная техника»

Руководитель:

Доктор педагогических наук, профессор

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.З. Власова

« \_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2017 г.

Автор работы:

Студент группы 2ИВТ 2 курса

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Е.Д. Стрижов

« \_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2019 г.

Санкт-Петербург

2019

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ ........................................................................................................... 3 ОСНОВНАЯ ЧАСТЬ ........................................................................................... 5

1. Теоретическая часть ....................................................................................... 5

1.1. Задача 1.................................................................................................... 5

1.2. Задача 2.................................................................................................... 8

2. Практическая часть ........................................................................................ 11 2.1. Моделирование тренажера по численным методам решения задач

линейной алгебры........................................................................................ 11

2.1.1. Решение СЛАУ методом треугольной факторизации: разложение

матрицы A на L – левую нижнюю и R – правую верхнюю треугольные матрицы .................................................................................................................11

2.1.2. Решение СЛАУ методом треугольной факторизации: разложение матрицы A на L – левую верхнюю и R – правую верхнюю треугольные матрицы .................................................................................................................12

3. Вывод ............................................................................................................ 14 ЗАКЛЮЧЕНИЕ .................................................................................................. 15 ЛИТЕРАТУРА .................................................................................................... 16 ПРИЛОЖЕНИЕ А .............................................................................................. 17 ПРИЛОЖЕНИЕ Б .............................................................................................. 18 ПРИЛОЖЕНИЕ В .............................................................................................. 20 ПРИЛОЖЕНИЕ Г .............................................................................................. 21

# ВВЕДЕНИЕ

В данной курсовой работе для написания модуля с методами треугольной факторизации, являющегося частью тренажера по численным методам решения задач линейной алгебры, разрабатываются вычислительные модели методов и программный код для их реализации и проведения необходимых расчетов.

Модель – это представление объекта или системы, она может быть, как точной копией, так и отображать некоторые характерные свойства объекта. Разработка компьютерных моделей актуальна как для учащихся и студентов, так и для людей, профессионально изучающих технологии компьютерного моделирования. Разработка вычислительной модели объекта и программного кода для проведения расчетов позволяет уменьшить рутинные вычисления и увеличить их точность, визуализировать результаты решения задач.

Цель данной курсовой работы заключается в разработке моделей методов треугольной факторизации и программного кода для решения систем линейных алгебраических уравнений (СЛАУ) данными методами.

Для достижения поставленной цели курсовой работы требуется решить следующие задачи:

* вывести формулы для двух вариантов треугольной факторизации:

1. Вариант разложения матрицы системы на левую нижнюю, находящуюся под главной диагональю, треугольные матрицы, и правую верхнюю, находящуюся над главной диагональю.
2. Вариант разложения матрицы системы на левую верхнюю, находящуюся над побочной диагональю, и правую верхнюю, находящуюся над. главной диагональю, треугольные матрицы.

* разработать программный код для решения СЛАУ по выведенным формулам
* создание интерфейса пользователя.

Объектом данного исследования являются технологии компьютерного моделирования.

Предметом исследования является разработка вычислительных моделей методов треугольной факторизации и программного кода для решения СЛАУ указанными методами.

Курсовая работа состоит из двух глав: теоретической - постановка задач и вывод рекуррентных формул и практической - разработка программного кода и интерфейса пользователя для решения поставленных задач.

# ОСНОВНАЯ ЧАСТЬ

# Теоретическая часть

Чтобы лучше понять термины, использованные в данной работе, обратимся к теории.

Матрица - математический объект, записываемый в виде прямоугольной таблицы элементов кольца или поля, которая представляет собой совокупность строк и столбцов, на пересечении которых находятся её элементы.

Система линейных алгебраических уравнений (СЛАУ) – система уравнений, каждое уравнение в которой является линейным — алгебраическим уравнением первой степени.

Факторизация (разложение) – это представление матрицы A в виде произведения нескольких матриц.

## Задача 1

Постановка задачи: решить систему системных алгебраических уравнений вида ([ПРИЛОЖЕНИЕ А](#bookmark)):

методом треугольной факторизации, используя структуру типа:

где L – левую нижняя, а R – правую верхняя. треугольные матрицы. Считается, что диагональные элементы матрицы R равны 1.

Вывод формул для решения: Изначально имеем СЛАУ:

(1)

в которой представим матрицу A в виде произведения двух треугольных матриц:

(2)

В этом случае решение СЛАУ сведется к последовательному решению двух треугольных систем с промежуточными неизвестными Z. Первоначально решается система относительно Z:

(3)

а затем решаем систему относительно Х:

(4)

Этап прямого хода

Найдем рекуррентные формулы для вычисления матриц L и R:

*i= 1/n*

(5)

*i= 1/n*

(6)

(7)

*i= 2/n*

(8)

1. Введем вспомогательный индекс t:

(9)

(10)

Результатом этапа прямого хода является получение рекуррентных формул для нахождения элементов матриц L и R.

Этап обратного хода:

Найдем корни СЛАУ. Обозначим

тогда

Для окончательного решения заданной СЛАУ необходимо найти рекуррентные формулы для вычисления Z и X:

(11)

(12)

(13)

(14)

Полученные формулы (5), (6), (7), (8), (9), (10), (11), (12), (13), (14) используем для моделирования тренажера по численным методам решения СЛАУ.

## Задача 2

Постановка задачи: решить систему системных алгебраических уравнений вида

методом треугольной факторизации, используя структуру типа

где L – левая верхняя, а R – правая верхняя треугольные матрицы. Считается, что диагональные элементы матрицы R равны 1.

Вывод формул для решения: Изначально имеем СЛАУ:

(15)

в которой представим матрицу A в виде произведения двух треугольных матриц:

(16)

В этом случае решение СЛАУ сведется к последовательному решению двух треугольных систем с промежуточными неизвестными Z. Первоначально решается система относительно Z:

(17)

а затем решаем систему относительно Х:

(18)

Этап прямого хода:

Найдем рекуррентные формулы для вычисления матриц L и R:

(19)

(20)

(21)

(22)

1. Введем вспомогательный индекс t:

(23)

(24)

Результатом этапа прямого хода является получение рекуррентных формул для нахождения элементов матриц L и R.

Этап обратного хода:

Найдем корни СЛАУ. Обозначим

тогда

Для окончательного решения заданной СЛАУ необходимо найти рекуррентные формулы для вычисления Z и X:

(25)

(26)

(27)

(28)

Полученные формулы (19), (20), (21), (22), (23), (24), (25), (26), (27), (28) используем для моделирования тренажера по численным методам решения СЛАУ.

# 2. Практическая часть

## 2.1. Моделирование тренажера по численным методам решения задач линейной алгебры

В данном разделе будет проводится разработка кода для реализации решения СЛАУ методом треугольной факторизации в тренажере по численным методам решения задач линейной алгебры.

Используемое оборудование: ПК, программное обеспечение “JetBrains Intellij IDE 2018.2.4”, язык программирования “Java”.

## 2.1.1. Решение СЛАУ методом треугольной факторизации: разложение

матрицы A на L – левую нижнюю и R – правую верхнюю треугольные матрицы

Введем функцию для решения СЛАУ методом треугольной факторизации:

ПРИЛОЖЕНИЕ Б

Введем функцию для устранения нулевых элементов на диагонали:

ПРИЛОЖЕНИЕ В

## 2.1.2. Решение СЛАУ методом треугольной факторизации: разложение матрицы A на L – левую верхнюю и R – правую верхнюю треугольные матрицы

Введем функцию для решения СЛАУ методом треугольной факторизации:

ПРИЛОЖЕНИЕ Г

Введем функцию для устранения нулевых элементов на диагонали:

ПРИЛОЖЕНИЕ В

# Вывод

С помощью выведенных формул и программного обеспечения “JetBrains IntellijIDE 2018.2.4” и библиотеки “PyQt5” был смоделирован тренажер по численным методам решения задач линейной алгебры. Результат работы отображен на снимках окон ([Рисунок 1](#bookmark1)) , ([Рисунок 2](#bookmark2))

# ЗАКЛЮЧЕНИЕ

При написании данной курсовой работы решались поставленные задачи посредством изучения различных способов их решения с помощью использования технологий компьютерного моделирования.

В результате выполненной работы был смоделирован тренажер по численным методам решения задач линейной алгебры, вывели формулы для решения СЛАУ методом треугольной факторизации, разработали программный код на языке программирования “Java”.

Получившееся приложение поможет пользователю в решении систем линейных алгебраических уравнений и различных других операциями над матрицами: умножение, транспонирование матриц и решение СЛАУ методом Гаусса.

# ЛИТЕРАТУРА

1. Акопов А. С. Компьютерное моделирование: учебник и практикум для среднего профессионального образования / А. С. Акопов. — Москва: Издательство Юрайт, 2019. — 389 с.
2. Булавин, Л.А., Выгорницкий Н.В., Лебовка Н.И. Компьютерное моделирование физических систем. - Долгопрудный: Издательский Дом «Интеллект», 2011. - 352 c.
3. Градов В.М. Компьютерное моделирование: Учебник / В.М. Градов, Г.В. Овечкин, П.В. Овечкин и др. - М.: Инфра-М, 2016. - 784 c.
4. Иванов А.П. ПРАКТИКУМ ПО ЧИСЛЕННЫМ МЕТОДАМ РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙПРАКТИКУМ ПО ЧИСЛЕННЫМ МЕТОДАМ РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ. - СПБ: СПБГУ, 2013. - 19 с.
5. Королев А.Л. Компьютерное моделирование / А.Л. Королев. - М.: Бином. ЛЗ, 2013. - 230 c.
6. Михеева Е. В. Практикум по информационным технологиям в профессиональной деятельности / Е. В. Михеева. – Академия: Москва, 2015. – 256 c.
7. Орлова И.В. Экономико-математические методы и модели: компьютерное моделирование: Учебное пособие / И.В. Орлова. - М.: Вузовский учебник, НИЦ Инфра-М, 2013. - 389 c.
8. Прохоренок Н., Дронов В. Python 3 и PyQt 5. Разработка приложений. - СПБ: БХВ-Петербург, 2016. - 832 с.
9. Сафонов В.И. Компьютерное моделирование: учебное пособие / В.И. Сафонов; Мордов. гос. пед. ин-т. – Саранск, 2009. – 92 с.
10. Сулейманов Р. Р. КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ МАТЕМАТИЧЕСКИХ ЗАДАЧ. УЧЕБНОЕ ПОСОБИЕ. - М.: БИНОМ. Лаборатория знаний, 2012. - 371 с.
11. Сирота А.А. Анализ и компьютерное моделирование информационных процессов и систем / Э.К. Алгазинов, А.А. Сирота; Под общ. ред. проф. д.т.н. Э.К. Алгазинов. - М.: ДИАЛОГ-МИФИ, 2009. - 416 c.
12. Тарасевич Ю.Ю. Математическое и компьютерное моделирование. Вводный курс: Учебное пособие / Ю.Ю. Тарасевич. - М.: ЛИБРОКОМ, 2013. - 152 c.
13. Тупицына А.И. Методы компьютерного моделирования физических процессов и сложных систем. Учебное пособие– СПб: Университет ИТМО, 2014. – 48 с.
14. Чичкарёв Е. А. Компьютерная математика с Maxima. Руководство для школьников и студентов / Е. А. Чичкарёв. – М. : ALT Linux, 2012. – 384 c.
15. Юрчук С.Ю. Численные методы. Решения систем линейных алгебраических уравнений: Учебно-методическое пособие. - М: МИСиС, 2018. - 98 с.

# ПРИЛОЖЕНИЕ А

A\*X=B в матричном виде:

=

# ПРИЛОЖЕНИЕ Б

float[][] matrixA;

float[][] matrixR;

float [][] matrixL;

float[] z;

float[] X;

float S;

double x;

int n, p;

Scanner in = new Scanner(System.in);

System.out.print("Input a number: ");

int a = in.nextInt();

matrixA = new float[a][a + 1];

matrixR = new float[matrixA.length][matrixA[0].length];

matrixL = new float[matrixA.length][matrixA[0].length];

for (int i =0; i < matrixA.length; i++){

for (int j = 0; j < matrixA[0].length; j++){

matrixA[i][j] = in.nextFloat();

}

}

in.close();

n = matrixL.length - 1;

for (int i = 0; i < matrixL.length; i++){

matrixL[i][0] = matrixA[i][0];

}

matrixR[0][0] = 1;

for (int j = 1; j < matrixL.length; j++){

matrixR[0][j] = matrixA[0][j]/(matrixL[0][0]);

}

for (int j = 1; j < matrixL.length; j++) {

matrixR[j][j] = 1;

for (int i = j; i < matrixR.length; i++) {

S = 0;

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

S += matrixL[i][k] \* matrixR[k][j];

}

matrixL[i][j] = matrixA[i][j] - S;

}

int i = j;

for (int t = j + 1; t < matrixR.length; t++) {

S = 0;

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

S += matrixL[i][k] \* matrixR[k][t];

}

matrixR[i][t] = (matrixA[i][t] - S) / matrixL[i][i];

}

}

z = new float[matrixA.length];

X = new float[matrixA.length];

z[0] = matrixA[0][matrixL[0].length-1]/matrixL[0][0];

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

S = 0;

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

S+= matrixL[i][k]\*z[k];

}

z[i] = (matrixA[i][matrixL[0].length-1] - S)/matrixL[i][i];

}

X[n] = z[n]/matrixR[n][n];

for (int i =n - 1; i > -1; i--){

S = 0;

for (int k = i+1; k <=n; k++){

S+= matrixR[i][k]\*X[k];

}

X[i] =( z[i] - S)/matrixR[i][i];

}

# ПРИЛОЖЕНИЕ В

private static void changeMatrix (int k, float[][] M) {

float x;

for (int j= 0; j <= M.length; j++){

x = M[k][j];

M[k][j] = M[M.length - k -1][j];

M[M.length - k - 1][j] = x;

}

}

# ПРИЛОЖЕНИЕ Г

float[][] matrixA;

float[][] matrixR;

float [][] matrixL;

float[] z;

float[] X;

float S;

double x;

int n, p;

Scanner in = new Scanner(System.*in*);

System.*out*.print("Input a number: ");

int a = in.nextInt();

matrixA = new float[a][a + 1];

matrixR = new float[matrixA.length][matrixA[0].length];

matrixL = new float[matrixA.length][matrixA[0].length];

for (int i =0; i < matrixA.length; i++){

for (int j = 0; j < matrixA[0].length; j++){

matrixA[i][j] = in.nextFloat();

}

}

in.close();

n = matrixL.length - 1;

for (int i = 0; i < matrixL.length; i++){

matrixL[i][0] = matrixA[i][0];

}

matrixR[0][0] = 1;

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

matrixR[0][j] = matrixA[n][j]/(matrixL[n][0]);

}

for (int j = n-1; j > -1; j--) {

p = n -j;

matrixR[p][p] = 1;

for (int i = j; i >-1; i--) {

S = 0;

for (int k = 0; k <= n-j; k++) {

S += matrixL[i][k] \* matrixR[k][p];

}

matrixL[i][p] = matrixA[i][p] - S;

}

int i = j;

for (int t = n - i + 1; t <= n; t++) {

S = 0;

for (int k = 0; k <= n-i; k++) {

S += matrixL[i][k] \* matrixR[k][t];

}

matrixR[p][t] = (matrixA[i][t] - S) / matrixL[i][p];

}

}

z = new float[matrixA.length];

X = new float[matrixA.length];

z[0] = matrixA[n][matrixL[0].length-1]/matrixL[n][0];

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

p = n -i;

S = 0;

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

S+= matrixL[p][k]\*z[k];

}

z[i] = (matrixA[p][matrixL[0].length-1] - S)/matrixL[p][i];

}

X[n] = z[z.length-1]/matrixR[n][n];

for (int i =n - 1; i > -1; i--){

S = 0;

for (int k = i+1; k <=n; k++){

S+= matrixR[i][k]\*X[k];

}

X[i] =( z[i] - S)/matrixR[i][i];

}