

MAT-MEX

C++

ПОЛНАЯ ДОКУМЕНТАЦИЯ К БИБЛИОТЕКЕ ПО РАБОТЕ С МАТРИЦАМИ И СЛУ.
ОФОРМЛЕНО ПРИ ПОМОЩИ L^AT_EX.

Студент: Шаламов Иван.

Группа: 151.

Санкт-Петербург
2017

Содержание

1	Вступление.	2
2	Препроцессорные директивы.	3
3	Функции и методы их использования.	4
4	Основные принципы работы.	6
5	Классификация возможных ошибок.	8
6	Использованные при создании библиотеки источники и материалы.	9
7	Заключение.	10

1 Вступление.

- Данная библиотека создавалась с целью ее дальнейшего использования в программном решении Систем Линейных Уравнений (далее – СЛУ).
- Основные объекты, фигурирующие в ее использовании – матрицы и вектора, составленные из вещественных чисел.
- Элементы матриц – коэффициенты при неизвестных в СЛУ.
- Элементы векторов – свободные члены СЛУ.
- Библиотека содержит несколько функций, позволяющих пользователю взаимодействовать с программой при решении СЛУ.

2 Препроцессорные директивы.

Препроцессорные директивы отвечают за подключение заголовочных файлов и определение некоторых констант. В данной библиотеке представлены следующие из них:

- **# define Matrix vector < vector <double> >** (Определение типа данных **vector** < **vector** <**double**> > одним словом **Matrix**).
- **# define Vector vector <double>** (Определение типа данных **vector** <**double**> одним словом **Vector**).
- **# include <iostream>** (Поток ввода-вывода).
- **# include <vector>** (Для использования **STL-контейнера Vector**).
- **# include <cstdlib>** (Для использования функции **exit(1)** при возникновении ошибок).
- **# include <iomanip>** (Для разметки данных (**setw**) при выводе их на экран в консоли).
- **# include <cmath>** (Для использования математических операций **sqrt** и **pow**).
- **# include <fstream>** (Поток ввода-вывода в/из файл/а).
- **# include <string>** (Для использования **STL-контейнера String**).

3 Функции и методы их использования.

БИБЛИОТЕКА СОДЕРЖИТ СЛЕДУЮЩИЕ ФУНКЦИИ:

- Для работы с LU-разложением:

ValueNLU – функция возвращения целочисленного значения n , аргументы – `int n`, возвращает введенное число.

SetMatrixLU – функция заполнения элементов матриц A, L, U, R числами, аргументы – `Matrix A, L, U, R, int n`, возвращаемое значение – `void`.

SetVectorLU – функция заполнения элементов векторов bLU, yLU, xLU числами, аргументы – `Vector bLU, yLU, xLU, int n`, возвращаемое значение – `void`.

CompositionLU – функция проверки корректности LU-разложения путем перемножения матриц L и U , аргументы – `Matrix L, U, R, int n`, возвращаемое значение – `void`.

ShowMatrixLU – функция вывода элементов матрицы на экран в консоли, аргументы – `Matrix A`, возвращаемое значение – `void`.

LU – функция реализации LU-разложения матрицы A , аргументы – `Matrix A, L, U, R, int n`, возвращаемое значение – `void`.

ShowVectorLU – функция вывода элементов вектора на экран в консоли, аргументы – `Vector xLU`, возвращаемое значение – `void`.

GaussLU – функция реализации метода Гаусса (LU) для решения СЛУ, аргументы – `Matrix A, L, U, R, Vector bLU, yLU, xLU, int n`, возвращаемое значение – `void`.

- Для работы с LLT-разложением:

ValueMC – функция возвращения целочисленного значения m , аргументы – `int m`, возвращает введенное число.

SetMatrixC – функция заполнения элементов матриц B, LC, LT, V числами, аргументы – `Matrix B, LC, LT, V, int m`, возвращаемое значение – `void`.

SetVectorC – функция заполнения элементов векторов bC, yC, xC числами, аргументы – `Vector bC, yC, xC, int m`, возвращаемое значение – `void`.

TransposeC – функция транспонирования матрицы, аргументы – `Matrix B, LT, int m`, возвращаемое значение – `void`.

CompositionC – функция проверки корректности LLT-разложения путем перемножения матриц L и L^T , аргументы – `Matrix LC, LT, V, int m`, возвращаемое значение – `void`.

ShowMatrixC – функция вывода элементов матрицы на экран в консоли, аргументы – `Matrix B`, возвращаемое значение – `void`.

Cholesky – функция реализации LLT-разложения матрицы B , аргументы – Matrix B , LC , LT , V , $\text{int } m$, возвращаемое значение – `void`.

ShowVectorC – функция вывода элементов вектора на экран в консоли, аргументы – Vector x_C , возвращаемое значение – `void`.

CholeskyLLT – функция реализации метода Холецкого (LLT) для решения СЛУ, аргументы – Matrix B , LC , LT , V , Vector b_C , y_C , x_C , $\text{int } m$, возвращаемое значение – `void`.

4 Основные принципы работы.

- Если Вам необходимо решить СЛУ, то Вы можете использовать данную библиотеку следующим образом:
 - 1 Обозначить Matrix A/B, L/LC, U/LT, R/V, Vector bLU/bC, yLU/yC, xLU/xC, int n/m.
 - 2 Использовать функцию **ValueNLU/ValueMC**, обозначив количество неизвестных переменных в системе.
 - 3 Использовать функцию **GaussLU/CholeskyLLT** для решения СЛУ при помощи метода Гаусса (LU)/Холецкого (LLT).
 - 4 Библиотека также рассчитана на одновременное использование обоих методов решения СЛУ.
- Библиотека рассчитана на то, что пользователь будет вызывать непосредственно функции **GaussLU** или **CholeskyLLT** для пошагового решения СЛУ, но также возможно использование имеющихся функций в любом другом порядке.
- Библиотека рассчитана на работу с пользователем при помощи инструкций, выводимых на экран в консоли, то есть Вам достаточно использовать какую-либо функцию и далее следовать предложенным инструкциям.
- Если при использовании функций заполнения матриц или векторов Вы хотели бы считать данные из файла, то необходимо сделать следующее:

- 1 В файл 1.txt ввести матрицу $N \times N$, элементами которой являются коэффициенты при неизвестных в исходной СЛУ (**необходимо** ввести именно столько элементов, сколько имеется коэффициентов в исходной СЛУ), если Вы хотите решить СЛУ методом Гаусса (LU), далее выбрать в консоли чтение из файла.
 - 2 В файл 2.txt ввести вектор, элементами которого являются свободные члены в исходной СЛУ (**необходимо** ввести именно столько элементов, сколько имеется свободных членов в исходной СЛУ), если Вы хотите решить СЛУ методом Гаусса (LU), далее выбрать в консоли чтение из файла.
 - 3 В файл 4.txt ввести матрицу $M \times M$, элементами которой являются коэффициенты при неизвестных в исходной СЛУ (**необходимо** ввести именно столько элементов, сколько имеется коэффициентов в исходной СЛУ), если Вы хотите решить СЛУ методом Холецкого (LLT), далее выбрать в консоли чтение из файла.
 - 4 В файл 5.txt ввести вектор, элементами которого являются свободные члены в исходной СЛУ (**необходимо** ввести именно столько элементов, сколько имеется свободных членов в исходной СЛУ), если Вы хотите решить СЛУ методом Холецкого (LLT), далее выбрать в консоли чтение из файла.
- Файл 3.txt предназначен для вывода в него ошибок, которые могут возникнуть при решении СЛУ методом Гаусса (LU) или методом Холецкого (LLT), подробнее о возможных ошибках см. в параграфе **КЛАССИФИКАЦИЯ ВОЗМОЖНЫХ ОШИБОК**.

5 Классификация возможных ошибок.

В ходе решения исходной СЛУ могут возникнуть различные ошибки, в библиотеке они отслеживаются через исключения и при возникновении записываются в файл 3.txt, а также выводятся на экран в консоли. Причины их возможного возникновения описаны в классификации ошибок:

- **Error 1:** It's impossible to realize the LU-decomposition, since there is a division by 0 (В ходе процесса LU-разложения было обнаружено деление на 0, исходная СЛУ не имеет решений).
- **Error 2:** It's impossible to realize the LLT-decomposition, since the proposed matrix isn't symmetric (Осуществить LLT-разложение невозможно, так как введенная Вами или считанная из файла матрица коэффициентов не является симметричной).
- **Error 3:** It's impossible to realize the LLT-decomposition, since the proposed matrix isn't positive-definite (Осуществить LLT-разложение невозможно, так как введенная Вами или считанная из файла матрица коэффициентов не является положительно определенной).
- **Error 4:** It's impossible to realize the LLT-decomposition, since there is a division by 0 (В ходе процесса LLT-разложения было обнаружено деление на 0, исходная СЛУ не имеет решений).
- **Error 5:** It's impossible to solve the system by the Gaussian Elimination (LU-decomposition)/Cholesky Decomposition (LLT-decomposition) method, since there is a division by 0 in the algorithm (В ходе решения исходной СЛУ методом Гаусса (LU)/Холецкого (LLT) было обнаружено деление на 0, исходная СЛУ не имеет решений).
- **Error 6:** Unable to open the file (Файла, который Вы пытаетесь открыть не существует).
- **Error 7:** Please, enter 1 or 2 to continue (Вы ввели не 1 или 2 для считывания из файла или ввода данных для решения исходной СЛУ).
- **Error 8:** Please, enter $n/m > 0$ to solve the system of linear equations (Вы ввели ≤ 0 значение количества неизвестных в исходной СЛУ).

6 **Использованные при создании библиотеки источники и материалы.**

- <http://ru.wikipedia.org/wiki/LU-разложение>.
- http://ru.wikipedia.org/wiki/Разложение_Холецкого.
- [http://algowiki-project.org/ru/Разложение_Холецкого_\(метод_квадратного_корня\)](http://algowiki-project.org/ru/Разложение_Холецкого_(метод_квадратного_корня)).
- [http://algowiki-project.org/ru/Метод_Холецкого_\(нахождение_симметричного_треугольного_разложения\)](http://algowiki-project.org/ru/Метод_Холецкого_(нахождение_симметричного_треугольного_разложения)).
- [http://algowiki-project.org/ru/Обратная_подстановка_\(вещественный_вариант\)](http://algowiki-project.org/ru/Обратная_подстановка_(вещественный_вариант)).
- [http://algowiki-project.org/ru/Прямая_подстановка_\(вещественный_вариант\)](http://algowiki-project.org/ru/Прямая_подстановка_(вещественный_вариант)).

7 Заключение.

Данная библиотека была собрана на базе CODE::BLOCKS 16.01.
Все права защищены.