Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский Нижегородский государственный университет им. Н. И. Лобачевского»

Факультет информационных технологий математики и механики

Отчёт по лабораторной работе

Структуры хранения матриц специального вида

Выполнил:

студент ф-та ИТММ гр. 38-03

Бозин Д.О

Проверил:

Заведующий лабораторией суперкомпьютерных технологий и высокопроизводительных вычислений

Лебедев И.Г.

Нижний Новгород

2022 г.

**Содержание**

[Введение 3](#_gjdgxs)

[Постановка задачи 4](#_30j0zll)

[Руководство пользователя 5](#_1fob9te)

[Руководство программиста 6](#_3znysh7)

[Описание структур данных 6](#_3znysh7)

[Описание алгоритмов 6](#_2et92p0)

[Описание структуры программы 6](#_17dp8vu)

[Заключение 7](#_3dy6vkm)

[Литература 8](#_3rdcrjn)

[Приложения 9](#_26in1rg)

[Приложение 1 9](#_lnxbz9)

[Приложение 2 9](#_2s8eyo1)

**Введение**

**C++** (читается *си-плюс-плюс*) — компилируемый, статически типизированный язык программирования общего назначения.

Поддерживает такие парадигмы программирования, как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование. Язык имеет богатую стандартную библиотеку, которая включает в себя распространенные контейнеры и алгоритмы, ввод-вывод, регулярные выражения, поддержку многопоточности и другие возможности. C++ сочетает свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником — языком C — наибольшее внимание уделено поддержке объектно-ориентированного и обобщенного программирования.

**C++** широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также игр. Существует множество реализаций языка C++, как бесплатных, так и коммерческих и для различных платформ. Например, на платформе x86 это GCC, Visual C++, Intel C++ Compiler, Embarcadero (Borland) C++ Builder и другие. C++ оказал огромное влияние на другие языки программирования, в первую очередь на Java и C#.

Синтаксис C++ унаследован от языка [C](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8_(%D1%8F%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)). Изначально одним из принципов разработки было сохранение совместимости с C. Тем не менее C++ не является в строгом смысле надмножеством C; множество программ, которые могут одинаково успешно транслироваться как компиляторами C, так и компиляторами C++, довольно велико, но не включает все возможные программы на C.

1. **Постановка задачи**

Написать классы для работы с векторами и матрицами, использовать шаблоны.

Вектора в математическом понимании: имеется набор значений из N мерного пространства, размерность задается как параметр.

Матрица должна быть наследником вектора.

Классы вектора и матрицы должны быть вынесены в статическую библиотеку.

Продемонстрировать их работу на примере (написать в main пример) и написать тесты.

Должны быть:

конструкторы (по умолчанию, инициализатор, копирования), деструктор, доступ к защищенным полям;

перегруженные операции: +,-,\*,/,=,==, [] потоковый ввод и вывод;

перегруженные операции +,-,\*,/ должны быть реализованы для векторов (вектор +-\*/ вектор), матриц (матрица +-\* матрица);

Дополнительное задание:

1 - транспонирование, вычисление определителя, ранга, обратной матрицы, проверка на подобие.

2 - приведения матрицы к треугольному виду, решение систем линейных уравнений.

3 - эффективные алгоритмы матричного умножения с использованием возможностей и принципов работы современных процессоров. Сравнение разных реализаций (проведение вычислительных экспериментов при больших порядках матриц).

**2. Руководство пользователя**

Для созданий вектора есть три конструктора: по умолчанию, инициализатор, копирования.

Доступ к полю len - количество элементов в векторе;.

Можно складывать, вычитать, умножать, делить, присваивать, сравнивать вектора и обращаться отдельно к элементам вектора.

Векторы можно выводить и задавать с клавиатуры. Если вы хотите ввести вектор с клавиатуры то сначала надо указать его длину.

Есть дополнительные методы для работы с векторами: изменение размера вектора, подсчет количества элементов определенного значения в векторе, нахождение всех индексов элементов, имеющих конкретное значение, 3 вида сортировки (пузырьком, вставками, быстрая).

Для созданий матрицы есть три конструкторов: по умолчанию, инициализатор (создает квадратную), копирования (копирует матрицу), пересылки.

Можно складывать, вычитать, умножать, присваивать матрицы к матрицам, сравнивать матрицы и обращаться отдельно к векторам матрицы.

Матрицы можно выводить и задавать с клавиатуры. Если вы хотите ввести матрицу с клавиатуры то сначала надо указать ее ширину, а затем длину.

Для созданий нижнетреугольной матрицы есть три конструкторов: по умолчанию, инициализатор, копирования (копирует матрицу), пересылки.

Можно складывать, вычитать, умножать, присваивать нижнегреугольные матрицы к нижнегреугольным матрицам, сравнивать нижнегреугольные матрицы и обращаться отдельно к векторам нижней треугольной матрицы.

**3. Руководство программиста**

***3.1 Описание структуры программ***

Заголовочный файл DynamicVector.h состоит из шаблонного класса TDynamicVector и реализации методов класса.

Заголовочный файл DynamicMatrix.h является наследником класса TDynamicVector содержит методы для работы с матрицами.

Заголовочный файл LowTriangleMatrix.h является наследником класса TDynamicVector содержит методы для работы с матрицами.

main - программа запрашивает выбор типа чисел: целые или дробные. При выборе соответствующего типа демонстрируются примеры методов.

***3.2 Описание структур данных***

Класса TDynamicVector имеет 2 поля sz и pMem, где находятся длинна массива и массив шаблонных элементов соответственно.

Класса TDynamicMatrix имеет 2 поля sz, pMem. pMem содержит вектор векторов, sz длина и ширина.

Класса TLowTriangleMatrix имеет 2 поля sz, pMem. pMem содержит вектор векторов, sz длина и ширина.

***3.3 Описание алгоритмов***

ВЕКТОР

Конструктор по умолчанию - конструктор инициализации со значением 1. Конструктор инициализатор принимает длину вектора, затем создает неинициализированный массив полученной длинны. Конструктор копирования копирует поля принимаемого вектора. Метод size возвращает значение поля sz. При выполнения операторов +, -, \*, / у 2 векторов сравнивается длина, если их длины равны то результатом будет вектор той же длины, полученный применением соответствующих операций поэлементно. Оператор присваивания вектора - копирует поля вектора. Сравнение векторов происходит поэлементно. Оператор [] принимает целочисленный индекс, если он не выходит за границы массива, то возвращает элемент находящийся на указанной позиции.

МАТРИЦА

Конструктор по умолчанию - конструктор инициализации со значением 1. Конструктор инициализатор создает квадратную матрицу (массив массивов). Конструктор копирования копирует поля полученной матрицы. Метод size возвращает значение поля sz. При выполнения операторов +, -, \* у 2 матриц сравниваются длина и ширина, если их длина и ширина равны то результатом будет матрица той же длины и ширины, полученная применением соответствующих операций поэлементно. Оператор присваивания матрицы - копирует поля матрицы. Сравнение матриц происходит поэлементно. Оператор [] принимает целочисленный индекс, если он не выходит за границы, то возвращает вектор находящийся на указанной позиции.

**4.Заключение**

Я сделал статическую библиотеку для работы с векторами и матрицами. Реализовал все необходимые методы.

**5. Эксперименты**

|  |  |  |  |
| --- | --- | --- | --- |
| Время в секундах | 1000 эл вектор, матр 100 на 100 | 10000 эл вектор, матр 1000 на 1000 | 10000000 эл вектор,  матр 10000 на 10000 |
| Вектор и вектор | 0.001 | 0.011 | 0.9 |
| Вектор и матрица | 0.001 | 0.02 | 2.6 |
| Матрица и Матрица | 0.001 | 16.437 | ~7.2 часов. |

# 6.Литература

1. <https://ru.wikipedia.org/wiki/C%2B%2B>.
2. https://github.com/LebedevIlyaG/3821

ааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааа

ааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааа

ааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааа

ааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааааа

**Приложения**

***Приложение 1***

***Приложение 2***