

# Лабораторная работа

## Задание 1.1

### Компьютерная обработка информации

#### Модели<sup>1</sup>

Все многообразие моделей делится на три класса.

##### 1. Материальные (натурные) модели:

- некие реальные предметы (макеты, муляжи, эталоны);
- уменьшенные или увеличенные копии, воспроизводящие внешний вид моделируемого объекта, его структуру (глобус, модель кристаллической решетки) или поведение (радиоуправляемая модель самолета, велотренажер).

##### 2. Абстрактные модели (геометрическая точка, математический маятник, идеальный газ, бесконечность).

3. **Информационные модели** – описание моделируемого объекта на одном из языков кодирования информации (словесное описание схемы, чертежи, карты, рисунки, научные формулы, программы и т. д.). Информационная модель, как и любой другой вид информации, должна иметь свой материальный носитель.

Одним из наиболее часто используемых типов информационных моделей является *таблица*, которая состоит из строк и столбцов.

#### Методы<sup>2</sup>

Точные методы позволяют найти точное решение системы уравнений за конечное число операций, если операции выполнялись без погрешностей. Итерационные методы позволяют найти некое приближение к точному решению; степень приближения результата применения итерационного метода к точному решению зависит от заданной погрешности вычислений. От нее же зависит количество операций, выполняемых при реализации итерационного метода. Рассмотрим методы, используемые для решения системы линейных алгебраических уравнений.

- **Метод Гаусса** или метод последовательного исключения переменных относится к классу точных методов. Он основан на приведении матрицы коэффициентов  $a_{ij}$  к треугольному виду.

---

1 [https://lib.susu.ru/ftd?base=SUSU\\_METHOD&key=000529228&dtype=F&etype=.pdf](https://lib.susu.ru/ftd?base=SUSU_METHOD&key=000529228&dtype=F&etype=.pdf) с.8

2 [https://lib.susu.ru/ftd?base=SUSU\\_METHOD&key=000529228&dtype=F&etype=.pdf](https://lib.susu.ru/ftd?base=SUSU_METHOD&key=000529228&dtype=F&etype=.pdf) с.28

- **Метод Крамера** — способ решения систем линейных алгебраических уравнений с числом уравнений равным числу неизвестных с ненулевым главным определителем матрицы коэффициентов системы.<sup>3</sup>
- **Метод Итераций** используется для решения систем уравнений, в том числе, линейных алгебраических. В последнем случае он гарантирует сходимость решения, если все элементы главной диагонали матрицы коэффициентов  $A$  не равны нулю.

## Средства<sup>4</sup>

Существуют различные системы классификации электронных средств обработки информации: по архитектуре, по производительности, по условиям эксплуатации, по количеству процессоров, по потребительским свойствам и т.д. Один из наиболее ранних методов классификации – классификация по производительности и характеру использования компьютеров. В соответствии с этой классификацией компьютерные средства обработки можно условно разделить на следующие классы:

- микрокомпьютеры;
- мэйнфреймы;
- суперкомпьютеры.

**Микрокомпьютеры.** Первоначально определяющим признаком микрокомпьютера служило наличие в нем микропроцессора, т.е. центрального процессора, выполненного в виде одной микросхемы. Сейчас микропроцессоры используются во всех без исключения классах ЭВМ, а к микрокомпьютерам относят более компактные в сравнении с мэйнфреймами ЭВМ, имеющие производительность до сотен МИПС (МИПС – миллион команд в секунду).

Современные модели микрокомпьютеров обладают несколькими микропроцессорами. Производительность компьютера определяется не только характеристиками применяемого микропроцессора, но и емкостью оперативной памяти, типами периферийных устройств, качеством конструктивных решений и др.

Микрокомпьютеры представляют собой инструменты для решения разнообразных сложных задач. Их микропроцессоры с каждым годом увеличивают мощность, а периферийные устройства – эффективность.

**Персональные компьютеры (ПК)** – это микрокомпьютеры универсального назначения, рассчитанные на одного пользователя и управляемые одним человеком. В класс персональных компьютеров входят различные вычислительные машины – от дешевых

---

3 [https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4\\_%D0%9A%D1%80%D0%B0%D0%BC%D0%B5%D1%80%D0%B0](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_%D0%9A%D1%80%D0%B0%D0%BC%D0%B5%D1%80%D0%B0)

4 [https://studopedia.ru/17\\_22619\\_klassifikatsiya-sredstv-obrabotki-informatsii.html](https://studopedia.ru/17_22619_klassifikatsiya-sredstv-obrabotki-informatsii.html)

домашних и игровых с небольшой оперативной памятью до сверхсложных машин с мощным процессором, винчестерским накопителем емкостью в десятки гигабайт, с цветными графическими устройствами высокого разрешения, средствами мультимедиа и другими дополнительными устройствами.

Персональные компьютеры можно классифицировать и по конструктивным особенностям. Они подразделяются на стационарные (настольные) и переносные. Переносные, в свою очередь, делятся на портативные (Laptop), блокноты (Notebook), карманные (Palmtop).

**Мэйнфреймы.** Предназначены для решения широкого класса научно-технических задач и являются сложными и дорогими машинами. Их целесообразно применять в больших системах при наличии не менее 200-300 рабочих мест. Несколько мэйнфреймов могут работать совместно под управлением одной операционной системы над выполнением единой задачи.

**Суперкомпьютеры.** Это очень мощные компьютеры с производительностью свыше 100 МФЛОПС (МФЛОПС – сто миллионов операций в секунду). Они называются сверхбыстродействующими. Создать такие высокопроизводительные ЭВМ по современной технологии на одном микропроцессоре не представляется возможным ввиду ограничения, обусловленного конечным значением скорости распространения электромагнитных волн, так как время распространения сигнала на расстояние нескольких миллиметров (линейный размер стороны микропроцессора) при быстродействии 100 млрд оп./с становится соизмеримым со временем выполнения одной операции. Поэтому суперЭВМ создаются в виде высокопараллельных многопроцессорных вычислительных систем.

## Структуры данных в компьютерной алгебре<sup>5</sup>

**Числовые данные:**<sup>6</sup>

- Целочисленные (shortint, integer, longint, byte, word, comp)
- Вещественные (real, single, double, extended)
- Символьные (char, string)
- Логические (Boolean)

**Структура данных** — это контейнер, который хранит данные в определенном макете. Этот «макет» позволяет структуре данных быть эффективной в некоторых операциях и неэффективной в других.

---

<sup>5</sup> <https://habr.com/ru/post/422259/>

<sup>6</sup> <http://window.edu.ru/resource/820/44820/files/KluchMatjash1.pdf>

## Какие бывают?

**Линейные**, элементы образуют последовательность или линейный список, обход узлов линеен. Примеры: Массивы. Связанный список, стеки и очереди.

**Нелинейные**, если обход узлов нелинейный, а данные не последовательны. Пример: граф и деревья.

## Основные структуры данных.

1. **Массивы** — это самая простая и широко используемая структура данных. Другие структуры данных, такие как стеки и очереди, являются производными от массивов. Каждому элементу данных присваивается положительное числовое значение (индекс), который соответствует позиции элемента в массиве. Большинство языков определяют начальный индекс массива как 0.
2. **Стеки** — абстрактный тип данных, представляющий собой список элементов, организованных по принципу LIFO (англ. last in — first out, «последним пришёл — первым вышел»). Это не массивы. Это очередь. [Придумал Алан Тьюринг](#). Примером стека может быть куча книг, расположенных в вертикальном порядке. Для того, чтобы получить книгу, которая где-то посередине, вам нужно будет удалить все книги, размещенные на ней. Так работает метод LIFO (Last In First Out). Функция «Отменить» в приложениях работает по LIFO.
3. **Очереди**. Подобно стекам, очередь — хранит элемент последовательным образом. Существенное отличие от стека – использование FIFO (First in First Out) вместо LIFO. Пример очереди – очередь людей. Последний занял последним и будешь, а первый первым ее и покинет.
4. **Связанные списки** - массив где каждый элемент является отдельным объектом и состоит из двух элементов – данных и ссылки на следующий узел. Принципиальным преимуществом перед массивом является структурная гибкость: порядок элементов связного списка может не совпадать с порядком расположения элементов данных в памяти компьютера, а порядок обхода списка всегда явно задаётся его внутренними связями.
5. **Графы** - набор узлов (вершин), которые соединены друг с другом в виде сети ребрами (дугами).
6. **Деревья** - иерархическая структура данных, состоящая из узлов (вершин) и ребер (дуг). Деревья по сути связанные графы без циклов.
7. **Префиксные деревья** - Разновидность дерева для строк, быстрый поиск. Словари. Т9.

8. **Хэш таблицы.** Хэширование — это процесс, используемый для уникальной идентификации объектов и хранения каждого объекта в заранее рассчитанном уникальном индексе (ключе). Объект хранится в виде пары «ключ-значение», а коллекция таких элементов называется «словарем». Каждый объект можно найти с помощью этого ключа. По сути это массив, в котором ключ представлен в виде хеш-функции.

## Системы компьютерной алгебры: достижения и перспективы

**Система компьютерной алгебры (СКА, англ. computer algebra system, CAS)** — это прикладная программа для символьных вычислений, то есть выполнения преобразований и работы с математическими выражениями в аналитической (символьной) форме.<sup>7</sup>

### Классификация систем компьютерной математики<sup>8</sup>

В настоящее время системы компьютерной математики (СКМ) можно разделить на семь основных классов: системы для численных расчётов, табличные процессоры, матричные системы, системы для статистических расчётов, системы для специальных расчётов, системы для аналитических расчётов (компьютерной алгебры), универсальные системы.

Каждая система компьютерной математики имеет нюансы в своей архитектуре или структуре. Тем не менее можно прийти к выводу, что у современных универсальных СКМ следующая типовая структура:

Центральное место занимает ядро системы — коды множества заранее откомпилированных функций и процедур, обеспечивающих достаточно представительный набор встроенных функций и операторов системы.

Интерфейс даёт пользователю возможность обращаться к ядру со своими запросами и получать результат решения на экране дисплея. Интерфейс современных СКМ основан на средствах популярных операционных систем Windows 95/98/NT и обеспечивает присущие им удобства работы.

Функции и процедуры, включённые в ядро, выполняются предельно быстро. Поэтому объём ядра ограничивают, но к нему добавляют библиотеки более редких процедур и функций.

Кардинальное расширение возможностей систем и их адаптация к решаемым конкретными пользователями задачам достигаются за счёт пакетов расширения систем. Эти пакеты (нередко и библиотеки) пишутся на собственном языке программирования той или иной СКМ, что делает возможным их подготовку обычными пользователями.

7 [https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0\\_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%BE%D0%B9\\_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D1%8B](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%BE%D0%B9_%D0%B0%D0%BB%D0%B3%D0%B5%D0%B1%D1%80%D1%8B)

8 <https://www.intuit.ru/studies/courses/3484/726/lecture/25605?page=3>

Ядро, библиотеки, пакеты расширения и справочная система современных СКМ аккумулируют знания в области математики, накопленные за тысячелетия её развития.

Возрастающий интерес к алгебраическим алгоритмам возник в результате осознания центральной роли алгоритмов в информатике. Их легко описать на формальном и строгом языке и с их помощью обеспечить решение задач, давно известных и изучавшихся на протяжении веков. В то время как традиционная алгебра имеет дело с конструктивными методами, компьютерная алгебра интересуется ещё и эффективностью, реализацией, а также аппаратными и программными аспектами таких алгоритмов. Оказалось, что при принятии решения об эффективности и определении производительности алгебраических методов требуются многие другие средства, например, теория рекурсивных функций, математическая логика, анализ и комбинаторика.

В начальный период применения вычислительных машин в символьной алгебре быстро стало очевидным, что непосредственные методы из учебников часто оказывались весьма неэффективными. Вместо обращения к методам численной аппроксимации компьютерная алгебра систематически изучает источники неэффективности и ведёт поиск иных алгебраических методов для улучшения или даже замены таких алгоритмов.

## Подборка СКА

### СКА Maxima<sup>9</sup>

**Maxima** — свободная система компьютерной алгебры, написанная на языке Common Lisp.

Произошла от системы Macsyma, разрабатывавшейся в Массачусетском технологическом институте с 1968 по 1982 годы в рамках проекта Project MAC, финансируемого Министерством энергетики США (DOE) и другими государственными организациями. Профессор Уильям Шелтер (англ. Bill Schelter) из Техасского университета в Остине поддерживал один из вариантов системы, известный как DOE Macsyma, с 1982 года до самой своей смерти в 2001 году. В 1998 году Шелтер получил от Министерства энергетики разрешение опубликовать исходный код DOE Macsyma под лицензией GPL, и в 2000 году он создал проект на SourceForge.net для поддержания и дальнейшего развития DOE Macsyma под именем Maxima.

Maxima имеет широкий набор средств для проведения аналитических вычислений, численных вычислений и построения графиков. По набору возможностей система близка к таким коммерческим системам, как Maple и Mathematica. В то же время она обладает высочайшей степенью переносимости: может работать на всех основных современных операционных системах на компьютерах, начиная от наладонных и вплоть до самых мощных.

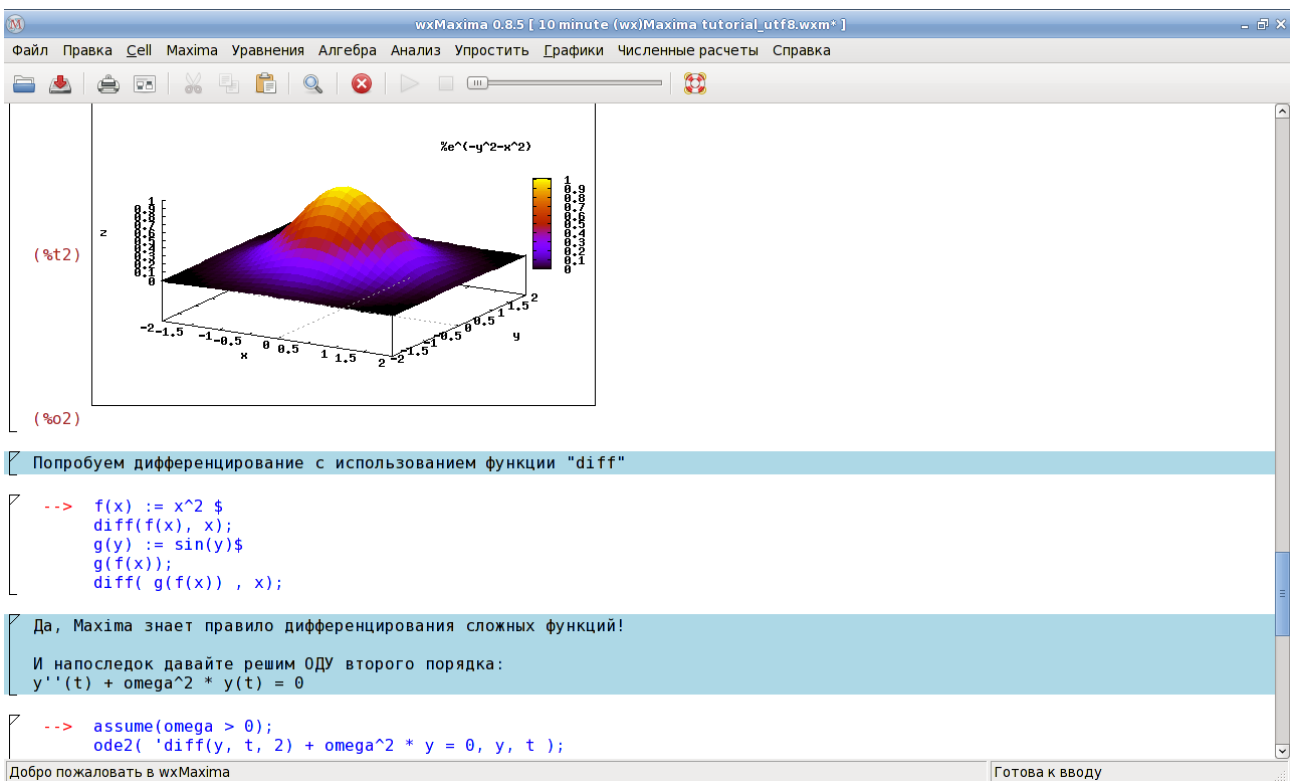
---

<sup>9</sup> <https://ru.wikipedia.org/wiki/Maxima>

Для системы построено несколько графических интерфейсов пользователя и надстроек: XMaxima (включен в поставку во многих ОС), wxMaxima (основан на wxWidgets) и других, а также может работать в режиме командной строки (используя псевдографику).

Для редактирования научных текстов в Maxima может использоваться программа texmacs, которая позволяет экспортировать документы в ряд популярных форматов, включая TeX/LaTeX и HTML/MathML.

Благодаря открытому коду системы появились производные решения, например, на основе Maxima сделана программа Stack, предназначенная для автоматизированной проверки правильности математических выражений, применяемая, в частности, для компьютерной проверки ответов обучающихся математике.



**Тип:** Система компьютерной алгебры (CAS)

**Автор:** Проект MAC Массачусетского технологического института

**Разработчик:** Уильям Шелтер, сообщество добровольцев

**Операционная система:** Linux, Mac OS X, Windows, FreeBSD, Android

**Сайт:** [maxima.sourceforge.net](http://maxima.sourceforge.net)

## СКА Matlab<sup>10</sup>

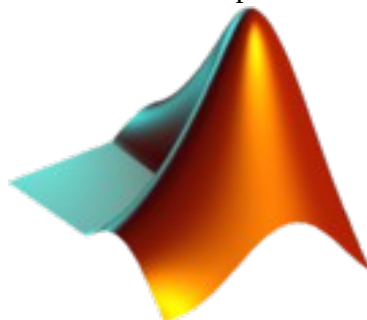
**MATLAB** (сокращение от англ. «Matrix Laboratory», в русском языке произносится как Матла́б) — пакет прикладных программ для решения задач технических вычислений. Пакет используют более миллиона инженерных и научных работников, он работает на большинстве современных операционных систем, включая Linux, Mac OS и Windows.

**Язык MATLAB** является высокоуровневым интерпретируемым языком программирования, включающим основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.

Программы, написанные на MATLAB, бывают двух типов — функции и скрипты. Функции имеют входные и выходные аргументы, а также собственное рабочее пространство для хранения промежуточных результатов вычислений и переменных. Скрипты же используют общее рабочее пространство. Как скрипты, так и функции сохраняются в виде текстовых файлов и компилируются в машинный код динамически. Существует также возможность сохранять так называемые pre-parsed программы — функции и скрипты, обработанные в вид, удобный для машинного исполнения. В общем случае такие программы выполняются быстрее обычных, особенно если функция содержит команды построения графиков.

Основной особенностью языка MATLAB являются его широкие возможности по работе с матрицами, которые создатели языка выразили в лозунге «думай векторно» (англ. Think vectorized).

<b>Тип</b>	<u><a href="#">математическое ПО</a></u>
<b>Разработчик</b>	<u><a href="#">The MathWorks</a></u> и <u><a href="#">Клеве Б. Молер</a></u>
<b><u>Написана на</u></b>	<u><a href="#">Си</a></u> , <u><a href="#">C++</a></u> , <u><a href="#">Фортран</a></u> и <u><a href="#">Java</a></u>
<b><u>Операционная система</u></b>	<u><a href="#">Microsoft Windows</a></u> , <u><a href="#">macOS</a></u> и <u><a href="#">GNU/Linux</a></u>
<b>Создаваемые форматы файлов</b>	<u><a href="#">Файл MATLAB-M</a></u> и <u><a href="#">Hierarchical Data Format</a></u>
<b><u>Лицензия</u></b>	проприетарная лицензия
<b>Сайт</b>	<a href="https://www.mathworks.com/products/matlab.html">https://www.mathworks.com/products/matlab.html</a>



<sup>10</sup> <https://ru.wikipedia.org/wiki/MATLAB>



## **Перспективы развития СКА**

В последние годы СКА набирают всё большую популярность и становятся доступны большему числу пользователей. Эта тенденция, очевидно, сохранится, и мы увидим гораздо более широкое внедрение СКА в образовательную среду и малый бизнес в ближайшем будущем. Повышаться также будет доступность СКА для обывателей.