

Федеральное агентство по образованию
Нижегородский государственный университет им. Н.И. Лобачевского

Национальный проект «Образование»
Инновационная образовательная программа ННГУ. Образовательно-научный центр
«Информационно-телекоммуникационные системы: физические основы и
математическое обеспечение»

А. В. Абросимов

Знакомство с математическими пакетами Maple V и Scientific Work Place

*Учебно-методические материалы по программе повышения
квалификации «Применение программных средств в научных
исследованиях и преподавании математики и механики»*

Нижний Новгород
2007

Учебно-методические материалы подготовлены в рамках
инновационной образовательной программы ННГУ: *Образовательно-
научный центр «Информационно-телекоммуникационные
системы: физические основы и математическое обеспечение»*

Абросимов А.В. Знакомство с математическими пакетами Maple V и Scientific Work Place.
Учебно-методический материал по программе повышения квалификации «Применение
программных средств в научных исследованиях и преподавании математики и механики е».
Нижний Новгород, 2007, 89 с.

Рассматриваются математические пакеты Maple и Scientific Work Place. В этих
пакетах делается обзор интерфейса: Описываются меню, панели инструментов.
Рассматриваются некоторые элементы языка, элементы программирования (для системы
Maple), наиболее важные символьные операции, описываются вычисления в этих системах и
построение графиков. Приводятся примеры.

© Абросимов А.В., 2007

ОГЛАВЛЕНИЕ

| | |
|---|----|
| Оглавление | 3 |
| Введение | 4 |
| 1 Часть 1. Система символьной математики Maple V. Классический интерфейс системы Maple | 5 |
| 2 Элементы языка Maple и элементы программирования. | 12 |
| 3 Вычисления и символьные операции в системе Maple | 32 |
| 4 Часть 2. Система Scientific Work Place. Описание интерфейса | 58 |
| 4 Вычисления и построение графиков | 78 |
| Заключение | 88 |
| Список литературы | 89 |

Знакомство с математическими пакетами Maple V и Scientific Work Place.

Введение.

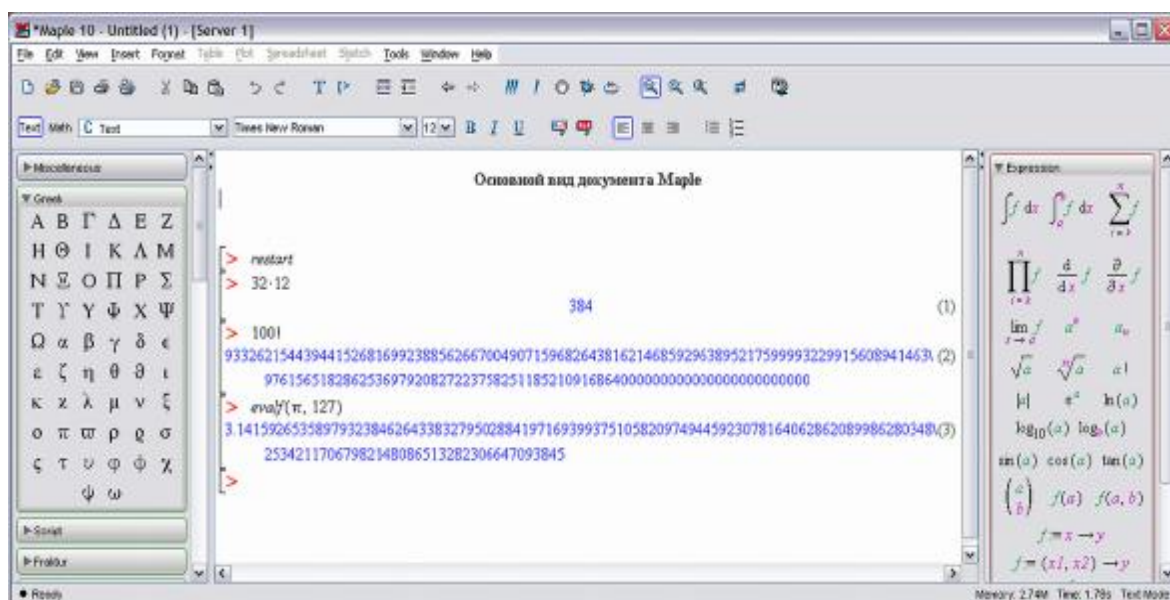
В последнее время появились различные интересные системы компьютерной алгебры, такие, как Axiom, Derive, Macsyma, Maple, Mathematica, Reduce, и др. Появились пакеты вычислительного характера (MathCad, Mathlab, ...).и пакеты подготовки научных результатов (Scientific Work Place, Math Office, ...). Здесь будут рассмотрены два из них: Maple V и SWP (Scientific Work Place).

Часть 1. Система символьной математики Maple V (Maple Waterloo).

Пакет Maple V является результатом совместного труда университета Ватерлоо (штат Онтарио, Канада) и Высшей технической школы (ETH, Цюрих, Швейцария). Этот пакет является лидером среди систем компьютерной алгебры и обеспечивает комфортную интеллектуальную среду для проведения математических исследований и оформления их результатов. Кроме того, символьный анализатор Maple включен в ряд других математических пакетов (MathCad, Matlab, SWP, ...) и, отчасти поэтому, эти пакеты достаточно популярны.

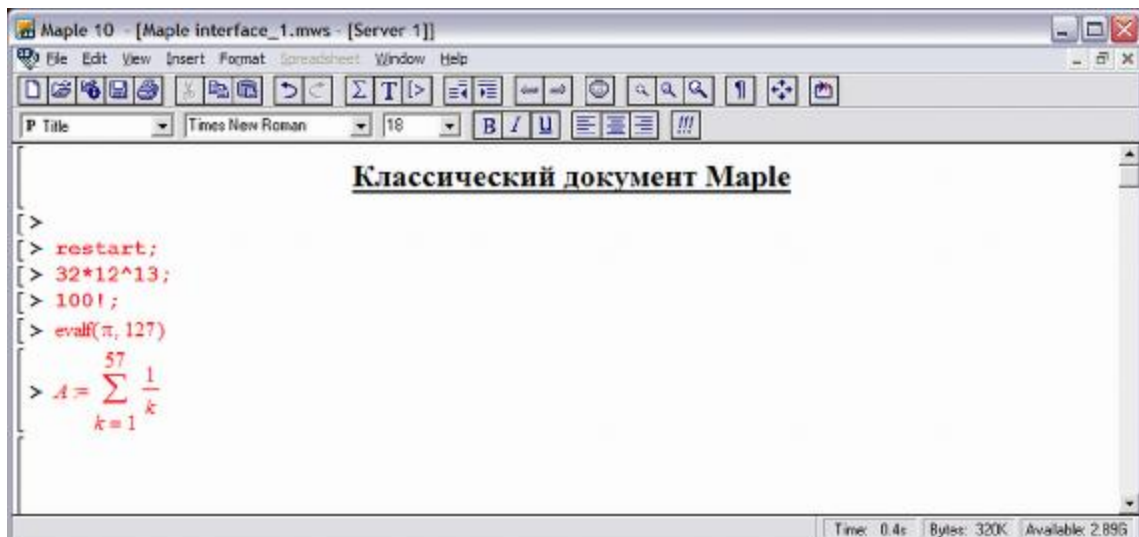
Maple состоит из ядра – процедур, написанных на языке C, библиотеки оптимизированных процедур, написанных на языке Maple, и интерфейса. Ядро выполняет большинство базовых операций. Библиотека содержит команды – процедуры, выполняемые в режиме интерпретации. Библиотеку можно пополнять своими собственными процедурами и, следовательно, расширять возможности Maple.

Интерфейс Maple в различных версиях может выглядеть различно. Так, например, в Maple 10 имеется два вида рабочего документа (worksheet). Первый документ, предлагаемый в качестве основного, имеет следующий вид.



Основной вид документа Maple.

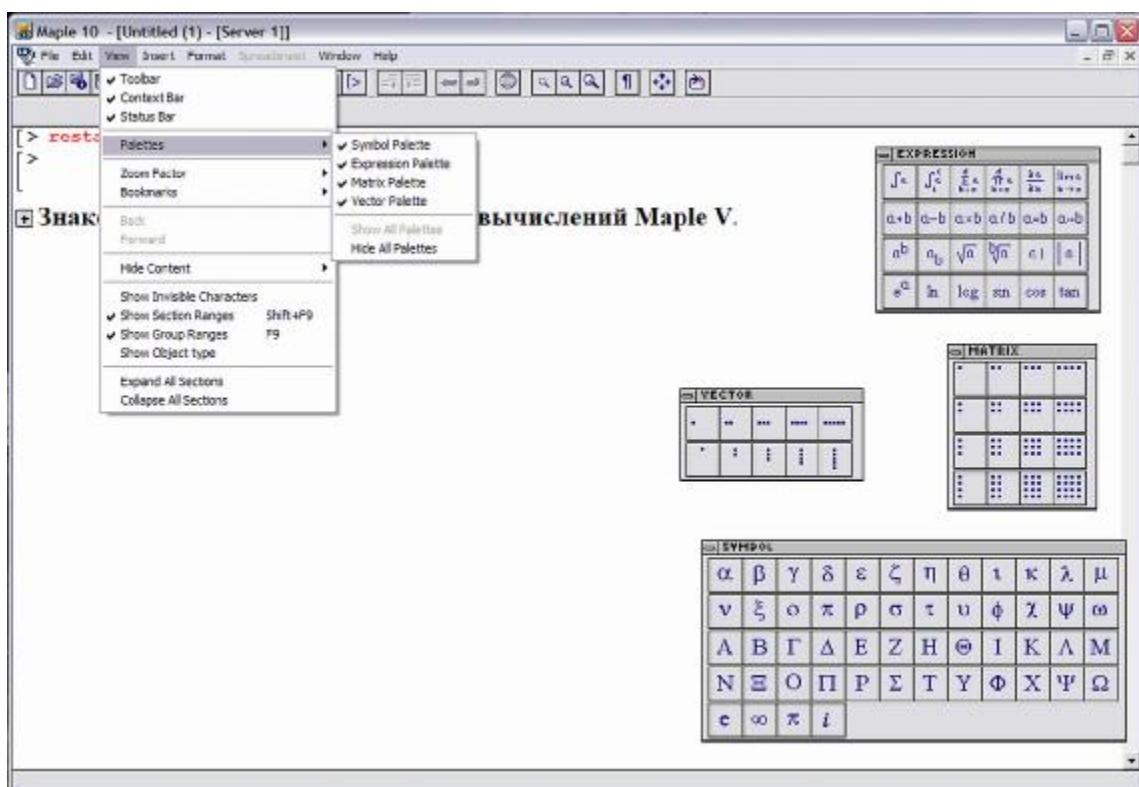
Вид второго документа (классический вид) не изменяется в последних версиях Maple, начиная с четвертой. При установке системы Maple 10 на рабочий стол выводятся два ярлыка с названиями Maple 10 и Classic Worksheet Maple 10. При вызове системы Maple с помощью второго ярлыка появляется классический вид документа. Он выглядит следующим образом.



Классический вид документа Maple.

Мы будем рассматривать здесь классический интерфейс.

Окно системы Maple V.



Как у всех приложений Windows интерфейс системы Maple имеет ряд характерных элементов. Эти элементы видны на рисунке 1.1. Перечислим их (сверху вниз).

- Строка состояния,
- строка главного меню
- главная панель инструментов,
- контекстная панель инструментов (ее вид зависит от режима работы),
- окно ввода и редактирования документа,
- строка состояния

Интерфейс Maple позволяет составлять документы, содержащие одновременно текст, команды Maple, которые можно преобразовать в привычную математическую форму, результаты вычислений в виде математических формул, таблицы и графические данные.

В основе интерфейса Maple лежит многооконный интерфейс Windows. Управление системой Maple возможно с помощью главного меню, панелей инструментов и палитр и сочетаний «горячих» клавиш. Многие действия можно выполнять с помощью мыши.

Справочная система Maple устроена по древообразному принципу и позволяет быстро найти интересующий раздел. Особенностью справочного раздела Maple является наличие там «живых» примеров, которые можно скопировать через буфер обмена в рабочий документ и отредактировать по своему усмотрению.

Меню системы Maple.

Главное меню системы Maple расположено сразу под строкой заголовка и предоставляет доступ к основным действиям. В главное меню входят следующие пункты.

File – работа с файлами, предварительный просмотр и печать документа,

Edit – редактирование документа и действия с буфером обмена,

View – управление видом окна,

Insert – операции вставки текста, команд, графики и т.д.

Format – форматирование документа,

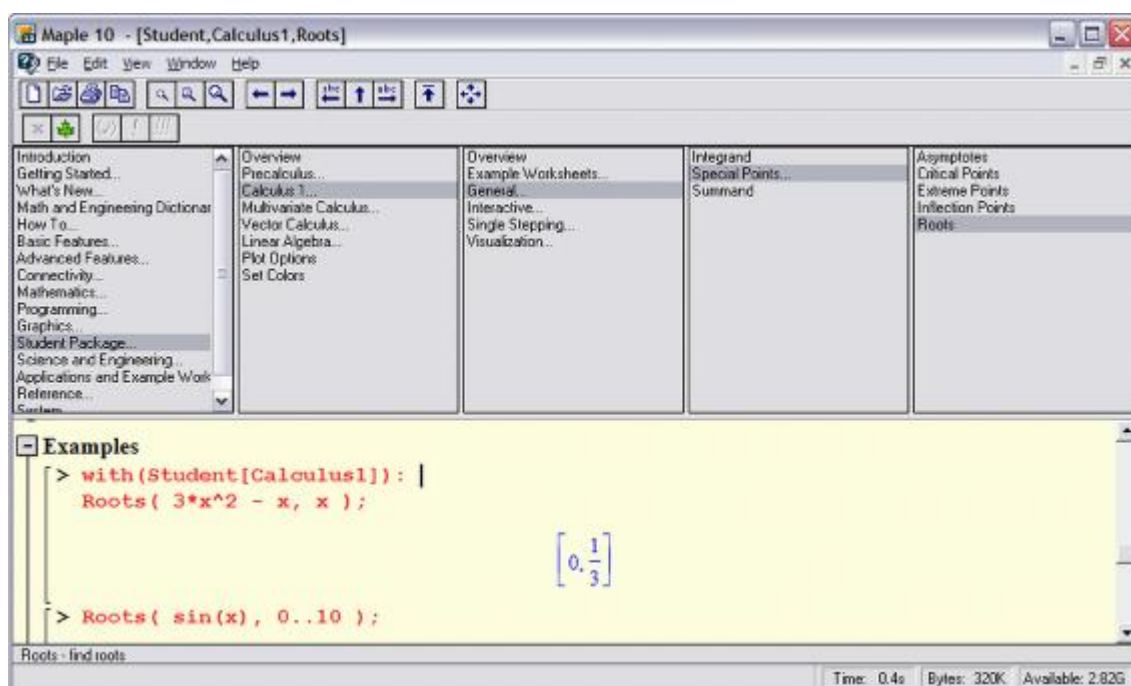
Spreadsheet – действия с таблицами,

Windows – управление оконным режимом,

Help – работа со справочным разделом.

Доступность пунктов главного меню зависит от состояния документа.

Назначение пунктов главного меню понятно большинству пользователей, работающих с Windows-приложениями. Поэтому отметим лишь особенности пункта **Help**. При вызове пункта **Help/Introduction** появляется окно справочной системы.

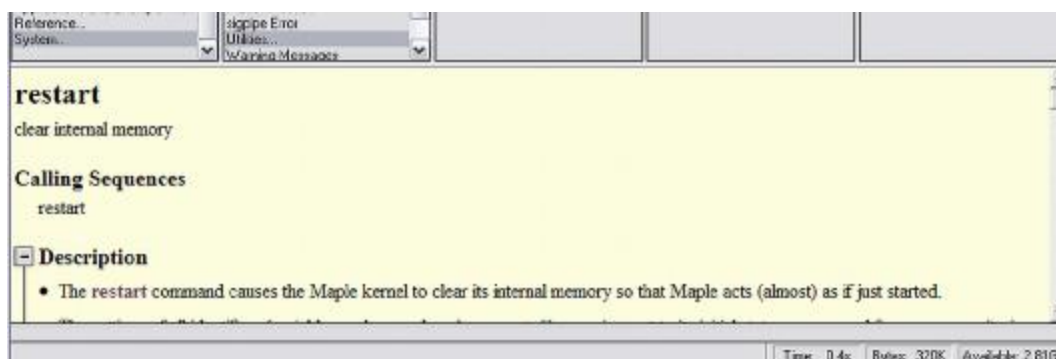


Эта система устроена по древообразному принципу: раздел/ подраздел/...и т.д. Главной особенностью этой системы является наличие «живых» примеров. Это означает, что пример действий по интересующему вопросу, приведенный в справочной системе через буфер обмена можно скопировать в свой рабочий документ и выполнить (или изменить по своему усмотрению и выполнить).

Справку по интересующему вопросу можно вызвать разными способами. Так, например, можно установить курсор в любое место интересующей команды и набрать






















сочетание клавиш Ctrl+F1. Тот же результат можно получить, поставив символ ? прямо перед интересующей командой.




Ø ?restart;



Главная панель инструментов.


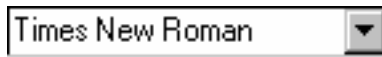






На панели инструментов находятся следующие кнопки

-  - Create a new worksheet (создать новый документ),
-  - Open an existing worksheet (открыть существующий документ),
-  - Open a specified URL (открыть URL),
-  Save the active worksheet (сохранить рабочий документ),
-  Print the active worksheet (печатать рабочий документ),
-  Cut the selection to the clipboard (вырезать выделенный фрагмент в буфер),
-  Copy the selection to the clipboard (скопировать выделенный фрагмент в буфер),
-  Paste the clipboard contents into the current worksheet (вставить из буфера),
-  Undo the last operation (отменить последнюю операцию),
-  Redo the previously undone operation (восстановить последнюю операцию),
-  Insert nonexecutable Standard Math in a text region (вставить (неисполняемую) формулу в область текста),
-  Insert text at the cursor (вставить текст в место перед курсором),
-  Insert a new execution group after the cursor (вставить исполняемую группу операторов после курсора),
-  Remove the section enclosing the selection (удалить выделенный раздел),
-  Enclose the selection in a section or subsection (вставить фрагмент в новый раздел),
-  Go backward in the hyperlink history (перейти назад по гиперссылке),
-  Go forward in the hyperlink history (перейти вперед по гиперссылке),
-  Cancel the computation in progress (остановить процесс вычислений),
-  Set the zoom magnification to 100% (установить масштаб дисплея 100%),
-  Set the zoom magnification to 150% (установить масштаб дисплея 150%),
-  Set the zoom magnification to 200% (установить масштаб дисплея 200%),

-  Toggle the display of nonprinting characters (изменение режима «невидимых» символов),
-  Resize the active window to fill the available space (переход окна в полный экран),
-  Restart Maple (перезагрузка Maple).

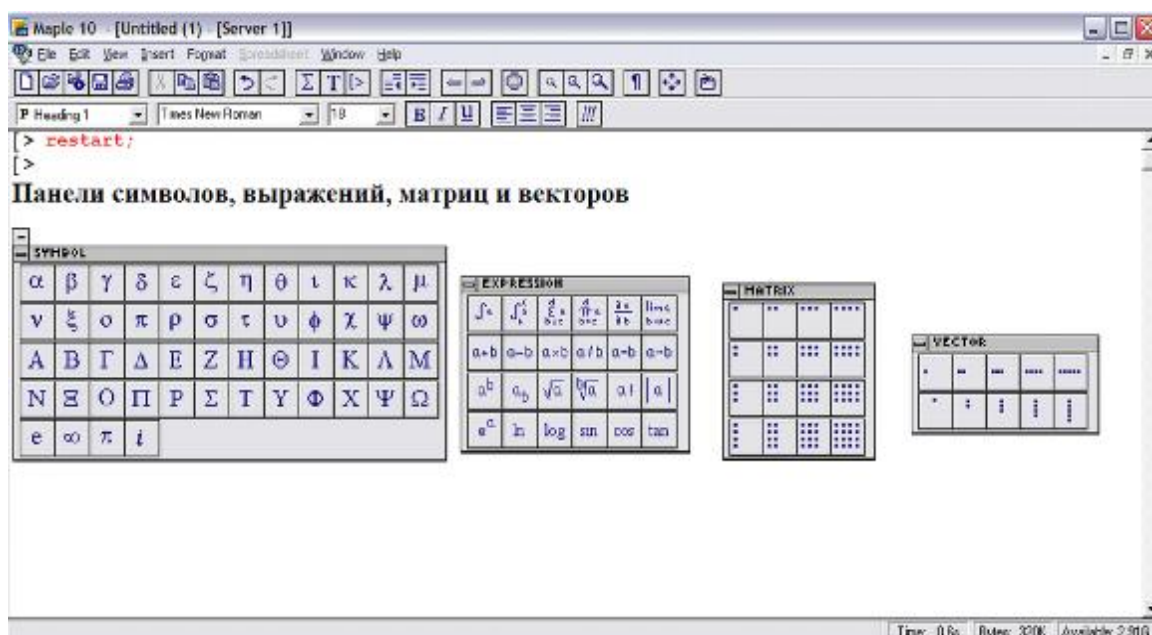
Панель форматирования текста (Context Bar for Text Regions)

Сюда входят следующие объекты.

-  Displays the text style (выбрать стиль текста),
-  Displays text font (выбрать шрифт),
-  Bold the selected text (выбрать жирный шрифт),
-  Italicize the selected text (выбрать курсив),
-  Underline the selected text (выбрать подчеркнутый шрифт),
-  Align the text on the left side (равнение по левому краю),
-  Center the text (равнение по центру),
-  Align the text on the right side (равнение по правому краю).

Палитры математических символов.

Для ввода математических символов, выражений, букв греческого алфавита, матриц и векторов можно (наряду с вводом с клавиатуры) можно пользоваться палитрами.



В меню **View** с помощью команды **Palettes** (палитры) можно вызвать следующие палитры математических символов:

Symbol Palettes – ввод букв греческого алфавита и некоторых математических символов,

Expression Palettes – ввод некоторых стандартных математических операторов и операций,

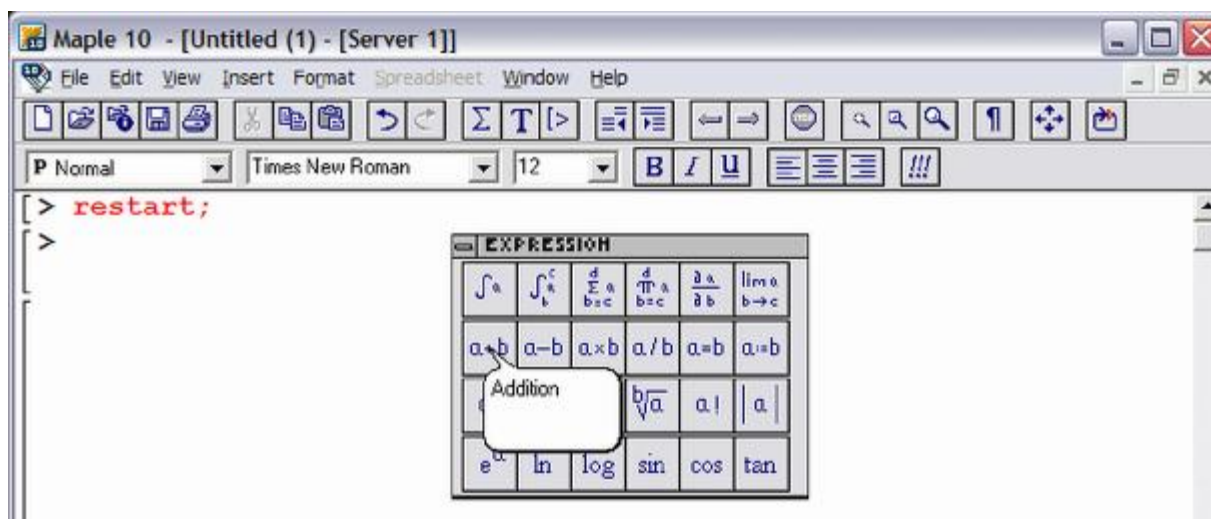
Matrix Palettes – ввод матриц разных размеров,

Vector Palettes – ввод векторов разных типов и размеров (вектор-столбец, вектор-строка).

Заметим, что вид вводимого выражения зависит от режима документа (**Standard Math** и **Maple Input**). Например, при вводе экспоненты в режиме **Standard Math** на дисплее отображается $e^?$, а в режиме **Maple Input** - **exp(%)**.

Всплывающие подсказки (balloon help).

Удобным элементом интерфейса являются всплывающие подсказки, появляющиеся при наведении курсора мыши на некоторые элементы окна. Они имеют вид облачка, в котором имеется краткое пояснение. Например, при наведении курсора на элемент **a+b** в палитре **Expression** появляется всплывающая подсказка – **addition** (сложение).



«Горячие» клавиши в системе Maple.

Многие действия в системе **Maple** можно выполнять с помощью зарезервированных сочетаний клавиш («горячие» клавиши). Они облегчают и ускоряют управление программой. Многие из них совпадают с сочетаниями, принятыми в Windows, MsOffice или другого приложения Windows.

Ниже приводятся некоторые сочетания клавиш.

«Горячие» клавиши для ввода текста и режима ввода.

- | | |
|--------------------------------------|--------|
| 1. Полужирное написание | Ctrl+B |
| 2. Курсив | Ctrl+I |
| 3. Подчеркнутое написание | Ctrl+U |
| 4. Включение/выключение ввода текста | F5 |
| 5. Включение режима ввода математики | Ctrl+M |
| 6. Включение режима ввода текста | Ctrl+T |

«Горячие» клавиши для выделения.

- | | |
|---|------------|
| 1. Выделение от курсора до начала строки | Shift+Home |
| 2. Выделение от курсора до конца строки | Shift+End |
| 3. Выделение от курсора до начала документа | Shift+PgUp |
| 4. Выделение от курсора до конца документа | Shift+PgDn |

«Горячие» клавиши для удаления, копирования и вставки.

- | | |
|---|----------|
| 1. Выделить все | Ctrl+A |
| 2. Копировать выделенное в буфер | Ctrl+C |
| 3. Вставить из буфера | Ctrl+V |
| 4. Удаление строки ввода (параграфа) | Ctrl+Del |
| 5. Вставка выражение в математической форме | Ctrl+R |
| 6. Вставка выражение в форме Maple | Ctrl+G |

«Горячие» клавиши переходов по документу.

- | | |
|---|--------------|
| 1. Переход к началу строки | Home |
| 2. Переход к концу строки | End |
| 3. Переход в начало документа | Ctrl+Home |
| 4. Переход в конец документа | Ctrl+End |
| 5. Создание новой строки | Shift+Enter |
| 6. Переход к следующей строке ввода | Tab |
| 7. Переход к предыдущей строке ввода | Shift+Tab |
| 8. Переход к справке по контексту | Ctrl+F1 |
| 9. Переход на предшествующий уровень вложенности разделов | Ctrl+UpArrow |

«Горячие» клавиши для работы с файлами.

- | | |
|--------------------------------|---------|
| 1. Создание нового документа | Ctrl+N |
| 2. Открытие документа | Ctrl+O |
| 3. Сохранение документа | Ctrl+S |
| 4. Печать документа | Ctrl+P |
| 5. Закрытие рабочего документа | Ctrl+F4 |
| 6. Завершение работы в Maple | Alt+F4 |


«Горячие» клавиши для просмотра документа


- | | |
|---------------------------------------|----------|
| 1. Перерисовка экрана (Redraw Screen) | Ctrl+L |
| 2. Просмотр групп ячеек | F9 |
| 3. Просмотр секций | Shift+F9 |
| 4. Разделение строки на две части | F3 |
| 5. Объединение смежных строк | F4 |
| 6. Деление раздела на две части | Shift+F3 |
| 7. Объединение смежных разделов | Shift+F4 |
| 8. Отмена предыдущей операции | Ctrl+Z |


«Горячие» клавиши для для установки масштаба.

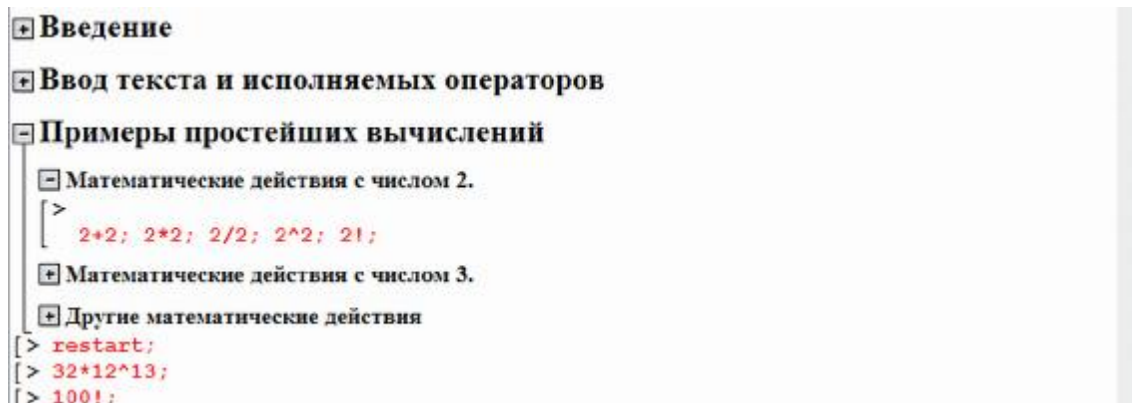
- | | |
|---------|--------|
| 1. 50% | Ctrl+1 |
| 2. 100% | Ctrl+2 |
| 3. 150% | Ctrl+3 |
| 4. 200% | Ctrl+4 |
| 5. 300% | Ctrl+5 |
| 6. 400% | Ctrl+6 |

Ввод информации.

Ввод информации происходит в разных режимах. Для ввода текстовой информации можно в главном меню выбрать команду **Insert/Text** или набрать сочетание **Ctrl+T** или нажать кнопку  на панели инструментов (см. **панель форматирования текста**).

Ввод каждой исполняемой группы операторов происходит с новой строки и начинается с символа **<**. В режиме ввода математики (режим по умолчанию) этот символ появляется автоматически при переходе на следующую строку. Его можно ввести из главного меню (**Insert/Maple Input**) и с панели инструментов (кнопка ). Ввод каждой исполняемой группы заканчивается символом **;** (точка с запятой).

В рабочем документе можно ввести структуру книги с помощью разделов (глав). Для создания раздела можно установить курсор в нужном месте и нажать кнопку . Внутри каждого раздела существует своя структура.



Элементы языка Maple и элементы программирования

Элементы языка Maple.

Язык Maple является входным языком общения с системой Maple и, одновременно, языком программирования. Большинство функций Maple написаны на этом языке.

Алфавит Maple содержит 26 малых латинских букв (от а до z) и 26 больших латинских букв (от А до Z), 10 арабских цифр (от 0 до 9) и 32 специальных символов (арифметические операторы, символ возведения в степень и др.).

К специальным одиночным и составным символам относятся, например, следующие элементы синтаксиса языка:

- % - системная переменная, хранящая результат предыдущей операции;
- : - фиксатор выражения, предотвращающий вывод результата в ячейку вывода;
- ; - фиксатор выражения, дающий вывод результата в ячейку вывода;
- #- указатель программного комментария;
- ' – ограничитель (например, 'string');
- \ - символ, имеющий множественные значения (см. Help/Backslash) и др.

Элементами языка Maple являются зарезервированные слова, использующиеся для создания условных выражений, циклов, процедур и управляющих команд. К ним относятся следующие: **and, break, by, catch, description, do, done, elif, else, end, error, export, fi, finally, for, from, global, if, in, intersect, local, minus, mod, module, next, not, od, option, options, or, proc, quit, read, return, save, stop, then, to, try, union, use, while.**

Простейшими объектами в Maple являются числа, константы, строки и имена. Числа могут быть целыми, рациональными, комплексными, корнями и числами с плавающей десятичной точкой. Например:

$$147, 12/9, \sqrt{2}, 2.718281828.$$

Операции с рациональными числами и корнями позволяют проводить абсолютно точные вычисления, так как отсутствует погрешность округления. Операции с вещественными числами проводятся по умолчанию с десятью значащими цифрами, но, переопределив зарезервированную константу **Digits**, можно работать с любой мантиссой. Предела для точности в Maple нет.

В Maple присутствуют все основные математические константы. Например:

Pi – число π ,

E – число e – основание натурального логарифма,

I – мнимая единица,
infinity – бесконечность,
Gamma – постоянная Эйлера,
True – логическая постоянная,

Имена этих переменных зарезервированы, а их значения не могут быть переопределены, в отличие от, например, управляющей постоянной **Digits**.

Строкой (**string**) является любой набор символов, заключенный в обратные кавычки. Например:

‘ Мой дядя правил самых честных, просто честных и, вообще, любых’.

Каждая переменная имеет имя – набор символов, начинающийся с буквы (большие и малые буквы различаются). В качестве переменных запрещено использовать зарезервированные слова. Поэтому, для безопасности, полезно перед введением нового имени **name**, ввести команду **?name**, чтобы узнать, свободно ли оно.

В Maple существует много типов переменных: от известных вещественного (**float**), целого (**integer**) и массива (**array**) до использующихся в символьных вычислениях: дробь (**fraction**), функция (**function**), процедура (**procedure**), множество (**set**) и т.д. Информацию о типе переменной можно получить с помощью команды **whattype**.

Используя переменные и знаки операций можно составлять выражения. Выражение в Maple есть объект, вполне соответствующий сути обычного математического выражения и может изменяться в соответствии с заданными правилами преобразований. Например, команда упрощения выражения **simplify** способна упрощать некоторые математические выражения, записанные в качестве ее параметра в круглых скобках.

> **simplify(sin(x)^2+cos(x)^2);**

1

Выражения и переменные обычно являются параметрами команд Maple. Обычная команда имеет вид:

command(p₁, p₂,...,p_n).

Здесь **command** - имя команды, а **p_i** - ее параметры. Часть команд вызываются автоматически, перед использованием других необходимо загрузить их в память командой **readlib**. Многие команды являются частью пакетов (библиотек), и до запуска команды пакет должен быть загружен командой **with(package)**, где **package** – имя пакета. В Maple имеются следующие пакеты.

algcures, ArrayTools, AudioTools, Cache, CodeGeneration, CodeTools, Combinat, Combstruct, ContextMenu, CurveFitting, DEtools, diffalg, diffforms, DiscreteTransforms, DocumentTools, Domains, ExternalCalling, FileTools, finance, GaussInt, genfunc, geom3d, geometry, gfun, Groebner, group, hashmset, ImageTools, InstallerBuilder, IntegerRelations, IntegrationTools, intrtrans, LargeExpressions, LibraryTools, liesymm, LinearAlgebra, LinearFunctionalSystems, LinearOperators, ListTools, Logic, LREtools, LREtools[HypergeometricTerm], Maplets, MathematicalFunctions, MathML, Matlab, MatrixPolynomialAlgebra, MmaTranslator, networks, numapprox, numtheory, optimization, OreTools, Ore_algebra, OrthogonalSeries, orthopoly, padic, PDEtools, plots, plottools, PolynomialIdeals, PolynomialTools, powseries, process, ProcessControl, QDifferenceEquations, RandomTools, RationalNormalForms, RealDomain, RegularChains, RootFinding, ScientificConstants, ScientificErrorAnalysis, simplex, Slode, SNAP, Sockets, SoftwareMetrics, SolveTools, Spread, Statistics, StringTools, Student, Student[Calculus1], Student[LinearAlgebra], Student[MultivariateCalculus], Student[Precalculus], Student[VectorCalculus]. Sumtools, SumTools, tensor, Tolerances, ToolboxInstaller[Data], Typesetting, TypeTools, Units, VariationalCalculus, VectorCalculus, Worksheet, XMLTools.

Названия следующих пакетов, известных по предыдущим версиям Maple, изменены :

codegen см. **CodeGeneration**, **linalg** см. **LinearAlgebra** и **VectorCalculus**, **stats** см. **Statistics**, **student** см. **Student**.

Например, в пакете **LinearAlgebra** доступны следующие команды: **RowDimension**, **RowOperation**, **RowSpace**, **ScalarMatrix**, **ScalarMultiply**, **ScalarVector**, **SchurForm**, **SingularValues**, **SmithForm**, **SubMatrix**, **SubVector**, **SumBasis**, **SylvesterMatrix**, **ToeplitzMatrix**, **Trace**, **Transpose**, **TridiagonalForm**, **UnitVector**, **VandermondeMatrix**, **VectorAdd**, **VectorAngle**, **VectorNorm**, **VectorScalarMultiply**, **ZeroMatrix**, **ZeroVector**. **Zip**.

Оценивание выражения.

Получая выражение, Maple оценивает его, т.е. устанавливает возможность его вычисления. Если выражение есть скалярная величина, то ее выражение будет выведено в ячейке вывода. Для переменных более сложных типов выводятся не их значения, а повторяются их имена. Имя возвращается и для неопределенных переменных.

Для оценивания выражений разных типов существуют свои операторы оценивания. Например, таковыми являются следующие.

>**eval(array)** – вычисление содержимого массива **array**,

>**evaf(expr,n)** – вычисление выражения **expr** и возвращение вычисленного значения в форме числа с плавающей точкой, имеющей **n** цифр после десятичной точки,

>**evalhf (expr)** – вычисление выражения **expr** и возвращение вычисленного значения с точностью, присущей данному компьютеру,

>**evaf(int(f,x=a..b))** – вычисление определенного интеграла $\int_a^b f dx$,

>**evalm(mexpr)** – вычисление матричного выражения **mexpr** и возвращение его,

>**evalc(cexpr)** – вычисление комплексного выражения **cmexpr** и возвращение его

Операторы.

Для конструирования выражений во входном языке Maple и языке программирования служат операторы.

Операторы обеспечивают определенные операции над данными, представленными операндами.

Существуют следующие типы операторов.

1. **binary** – бинарные операторы (два операнда),
2. **unary** – унарные операторы (один операнд),
3. **nullary** – нульарные операторы (без операндов – одна, две и три пары кавычек),
4. **precedence** – операторы старшинства (включая логические операторы),
5. **functional** – функциональные операторы.

К бинарным операторам относятся:

+ – сложение,

- – вычитание,

***** – умножение,

/ – деление,

^, или ****** – возведение в степень,

mod – остаток от деления,

\$ – оператор последовательности,

@ – оператор композиции,

. – разделительная точка,

@@ – повторение композиции,

' – разделитель выражений,

:= – присваивание,

.. – задание интервала,

&* – некоммутативное умножение,

<string> – нейтральный оператор,

\parallel - конкатенация,
 $<$ - меньше,
 $<=$ - меньше или равно,
 $>$ - больше,
 $>=$ - больше или равно,
 $=$ - равно,
 $<>$ - неравно,
and – логическое «и»,
or – логическое «или», и т.д.

Примеры.

$> a + b; 2 + 3;$

$a + b$

5

$> a - b; 2 - 3;$

$a - b$

-1

$> a * b; 2 * 3;$

$a b$

6

$> a / b; 2 / 3;$

$\frac{a}{b}$

$\frac{2}{3}$

$> a ^ b; 2 ^ 3;$

a^b

8

$> a \text{ mod } b;$

$\text{modp}(a, b)$

$> a < b;$

$a < b$

$> a <= b;$

$a \leq b$

$> a > b;$

$b < a$

$> a >= b;$

$b \leq a$

$> a = b;$

$a = b$

$> a <> b;$

$a \neq b$

$> a -> b;$

| | |
|---|-----------------------------------|
| | $a \rightarrow b$ |
| > a \$ 3; | a, a, a |
| > a@b; | $a@b$ |
| > a@@n; f @@ 2; | $a^{(n)}$ $f^{(2)}$ |
| > a b; "foo" "bar"; | ab "foobar" |
| > a . b; | $a . b$ |
| > a .. b; | $a .. b$ |
| > a :: b; | $a::b$ |
| > a and b; true and false; | $a \text{ and } b$ $false$ |
| > a or b; true or false; | $a \text{ or } b$ $true$ |
| > a xor b; true xor false; | $a \text{ xor } b$ $true$ |
| > a implies b; true implies false; | $a \text{ implies } b$ $false$ |
| > a subset b; { 2, 3 } subset { 1, 2, 3 }; | $a \subseteq b$ $true$ |
| > a union b; { 1, 2 } union { 2, 3 }; | $b \cup a$ $\{1, 2, 3\}$ |
| > a intersect b; { 1, 2 } intersect { 2, 3 }; | $b \cap a$ $\{2\}$ |


```

> a minus b; { 1, 2 } minus { 2, 3 };
      a \ b
      { 1 }

> `&x` := (x,y) -> 2 * x + 3 * y; a &x b;
      &x := (x, y) → 2 x + 3 y
      2 a + 3 b

> a , b;
      a, b

> a := b; a;
      a := b
      b

```

К унарным операторам относятся следующие операторы.

+ - унарный плюс (префикс),
 - - унарный минус (префикс),
 ! – факториал (префикс),
 Not – логическое отрицание (префикс),
 . – десятичная точка (префикс),
 \$ - последовательность (префикс),
 & name – нейтральный оператор, метка (префикс).

Примеры.

```

> +3; -3;
      3
      -3

> 8!;
      40320

> +3.;
      3.

> .3;
      0.3

> $ 1 .. 10;
      1, 2, 3, 4, 5, 6, 7, 8, 9, 10

```

Функциональные операторы.

Функциональные операторы являются альтернативами функций и записываются в двух формах – в виде стрелки или в виде угловых скобок:

1. vars -> result,
2. < vars | result >.

Эти операторы могут использоваться для реализации подстановок. Например, запись $x \rightarrow x^3$ означает подстановку x^3 на место переменной x . Возможны подстановки в множественной форме: $(x, y) \rightarrow x^2 + y^2$, $(x, y, z) \rightarrow (x*y, y*z)$, и т.д.

Примеры.

```

> f := x -> 3*x + 5;
      f := x → 3 x + 5

> f(2);
      11

```

```
> g := (x,y) -> sin(x)*cos(y) + x*y;
g := (x, y) → sin(x) cos(y) + x y

> g(Pi/2, Pi);
-1 + 1/2 π²

> h := x -> (2*x, x^3);
h := x → (2 x, x³)

> h(3);
6, 27
```

Математические функции.

Maple имеет полный набор элементарных математических функций. Все они имеют один аргумент. Этот аргумент может быть целым, рациональным, вещественным или комплексным числом. В ответ на обращение к ним элементарные функции возвращают соответствующее значение. Если известно имя нужной функции, то обращение к ней происходит с помощью обращения к этому имени.

Например, при вычислении выражения e^π и присваивании результата переменной **A**, достаточно вызвать команду

```
> A:=exp(Pi); (в режиме Maple Input), или
> A:=e^pi (в режиме Standard Math).
```

Опишем соответствие математической записи некоторых функций с записью в Maple.

e^x – exp(x),
 $\ln x$ – ln(x),
 $\log_{10} x$ – log10(x),
 \sqrt{x} – sqrt(x),
 $|x|$ – abs(x),
 $\operatorname{sgn} x$ – signum(x).

К целочисленным функциям относятся, например, следующие функции:

factorial(n) – вычисление факториала (можно использовать оператор !),
iquo(a,b) – целочисленное деление a на b,
irem(a,b) – остаток от деления a на b,
igcd(a,b) – наибольший общий делитель,
ilcm(a,b) – наименьшее общее кратное.

Примеры.

```
> irem(23,4,'q');
3

> q;
5

> iquo(23,4,'r');
5

> r;
3

> irem(-23,4);
-3

> irem(23,-4);
3
```

```

> irem(-23,-4);
-3

> igcd();
0

> ilcm();
1

> igcd( -10, 6, -8 );
2

> ilcm( -10, 6, -8 );
120

```

В Maple определены следующие тригонометрические функции.

sin – синус,
cos – косинус,
tan – тангенс,
cot – котангенс,
sec – секанс,
csc – cosecant.

Примеры.

```

> sin(0);
0

> cos(Pi);
-1

> expand(sin(x+y));
sin(x) cos(y) + cos(x) sin(y)

> combine(%,trig);
sin(x + y)

```

К обратным тригонометрическим функциям относятся следующие функции.

arcsin – арксинус,
arccos – арккосинус,
arctan – арктангенс,
arccot – арккотангенс,
arcsec – арксеканс,
arccsc – арккосеканс.

Примеры.

```

> sin(arccos(x));
 $\sqrt{1-x^2}$ 

```

Гиперболические функции и обратные к ним представлены следующим образом.

sinh – гиперболический синус,
cosh – гиперболический косинус,
tanh – гиперболический тангенс,
coth – гиперболический котангенс,
sech – гиперболический секанс,
csch – гиперболический cosecant,
arcsinh – гиперболический арксинус,
arccosh – гиперболический арккосинус,

$\operatorname{arctanh}$ – гиперболический арктангенс,
 $\operatorname{arccoth}$ – гиперболический арккотангенс
 $\operatorname{arcsech}$ – гиперболический арксеканс,
 $\operatorname{arccsch}$ – гиперболический арккосеканс.

Примеры.

> **arcsech(1);**

0

> **arcsinh(1.2+3.4*I);**

1.960545624+ 1.218868917I

> **D(arcsech);**

$$z \rightarrow -\frac{1}{z^2 \sqrt{\frac{1}{z}-1} \sqrt{\frac{1}{z}+1}}$$

> **diff(arcsech(x),x);**

$$-\frac{1}{x^2 \sqrt{\frac{1}{x}-1} \sqrt{\frac{1}{x}+1}}$$

> **convert(arccosh(x),ln);**

$$\ln(x + \sqrt{x-1} \sqrt{x+1})$$

К степенным и логарифмическим функциям в Maple относятся следующие функции.

\exp – показательная функция (экспонента),

$\operatorname{ilog10}$ – целая часть логарифма по основанию 10,

ilog – целая часть натурального логарифма,

\ln – натуральный логарифм,

$\log[b]$ – логарифм по заданному основанию b (библиотечная функция),

$\log10$ –

sqrt – квадратный корень,

Примеры.

> **ln(1);**

0

> **diff(ln(x),x);**

$$\frac{1}{x}$$

> **ln(3.14+2.71*I);**

1.422562238+ 0.7120258406I

> **ln(3+4*I);**

$$\ln(3 + 4I)$$

> **evalc(%);**

$$\ln(5) + \arctan\left(\frac{4}{3}\right)I$$

> **log[2](exp(1));**

$$\frac{1}{\ln(2)}$$

> **log10(10000);**

```

4
> simplify(%);
4
> exp(-1);

$$e^{(-1)}$$

> evalf(%);
0.3678794412
> exp(1.379);
3.970928713
> diff(exp(-x),x);

$$-e^{(-x)}$$

> evalf(exp(1));
2.718281828
> exp(I*Pi)+1;
0
> exp(1.234+5.678*I);
2.824884809- 1.954188170I
> evalc(exp(x+I*y));

$$e^x \cos(y) + e^x \sin(y) I$$

> sqrt(3.0);
1.732050808
> sqrt(3);

$$\sqrt{3}$$

> sqrt(4);
2
> sqrt(12);

$$2\sqrt{3}$$

> sqrt(-4);

$$2I$$

> sqrt(3+4*I);

$$2+I$$

> sqrt(4+2*sqrt(3));

$$\sqrt{3}+1$$

> sqrt(x);

$$\sqrt{x}$$

> sqrt(-9*x^2*y);

$$3\sqrt{-x^2 y}$$

> sqrt(-9*x^2*y,symbolic);

$$3x\sqrt{-y}$$


```

```
> assume(x>0);
sqrt(-9*x^2*y);
```

$$3x\sqrt{-y}$$

```
> assume(x<0);
sqrt(-9*x^2*y);
```

$$-3x\sqrt{-y}$$

```
> sqrt(-9*y/x^5);
```

$$3\frac{\sqrt{-\frac{y}{x}}}{x^2}$$

```
> f := (x-1)^2*y;
```

$$f := (x - 1)^2 y$$

```
> sqrt(f);
```

$$(-x + 1)\sqrt{y}$$

```
> sqrt(expand(f));
```

$$\sqrt{x^2 y - 2xy + y}$$

Функции с элементами сравнения.

abs – абсолютное значение (модуль) числа,
 ceil – наименьшее целое, не меньшее аргумента,
 floor – наибольшее целое, не превосходящее аргумента,
 frac – дробная часть,
 trunk – целое округленное, ближайшее к нулю,
 round – целое округленное (без дробной части),
 signum – сигнум (известная функция «знак»).

Примеры.

```
> abs(-11);
```

$$11$$

```
> abs(3-4*I);
```

$$5$$

```
> abs(cos(3));
```

$$-\cos(3)$$

```
> a:=abs(2*x-3);
```

$$a := |2x - 3|$$

```
> x:=1;
```

$$x := 1$$

```
> a;
```

$$1$$

```
> abs(sqrt(2)*I*u^2*v);
```

$$\sqrt{2}|u^2 v|$$

```
> expand(%);
```

$$\sqrt{2}|u|^2|v|$$

```
> combine(%,abs);
```

$$\sqrt{2}|u^2 v|$$

```

> diff(abs(y),y);
                                abs(1,y)
> diff(abs(y),y,y);
                                signum(1,y)
> abs(2,y);
                                signum(1,y)
> trunc(7);
                                7
> trunc(8/3);
                                2
> trunc(-2.4);
                                -2
> trunc(Pi);
                                3
> trunc(x);
                                trunc(x)
> trunc(3.5+4.2*I);
                                3+4 I
> round(8/3);
                                3
> frac(8/3);
                                 $\frac{2}{3}$ 
> floor(8/3);
                                2
> floor(-2.4);
                                -3
> floor(2.7+3.5*I);
                                3+3 I
> diff(floor(x),x);
                                floor(1,x)
> floor(1,3.5);
                                0

```

Функции комплексного аргумента.

Для комплексных чисел и данных, кроме уже перечисленных функций определены следующие функции.

argument – аргумент комплексного числа,

conjugate – комплексно-сопряженное число,

Im – мнимая часть числа,

Re – вещественная часть числа,

Polar – полярное представление комплексного числа (библиотечная функция),

Примеры.

```

> conjugate(3+5*I);

```

| | |
|---|---|
| | $3 - 5I$ |
| <code>> conjugate((3+5*I)*z);</code> | $(3 - 5I) \bar{z}$ |
| <code>> conjugate(sin(exp(1)));</code> | $\sin(e)$ |
| <code>> conjugate(exp(3*I));</code> | $\frac{1}{e^{(3I)}}$ |
| <code>> conjugate(polar(3,Pi/7));</code> | $\text{polar}\left(3, -\frac{1}{7}\pi\right)$ |
| <code>> Re(x);</code> | $\Re(x)$ |
| <code>> Im(x*y);</code> | $\Im(xy)$ |
| <code>> assume(z, real);</code> <code>Re(x*y+z);</code> | $z \sim + \Re(xy)$ |
| <code>> Re(x*z);</code> | $z \sim \Re(x)$ |
| <code>> Re(Pi+I*exp(1));</code> | π |
| <code>> Re(cosh(3+4*I));</code> | $\cosh(3) \cos(4)$ |
| <code>> Im(exp(I));</code> | $\sin(1)$ |
| <code>> Im(ln(-1));</code> | π |
| <code>> Im(polar(3,Pi/7));</code> | $3 \sin\left(\frac{1}{7}\pi\right)$ |
| <code>> argument(abs(z) * exp(I * 2*Pi/3));</code> | $\frac{2}{3}\pi$ |
| <code>> argument(polar(2, Pi/7));</code> | $\frac{1}{7}\pi$ |

Специальные математические функции.

Ниже приводятся наиболее важные специальные функции, имеющиеся в Maple.
 AiryAi (Bi), AiryAi (Bi) Zeros – функции Эйри,
 AngerJ – функция Ангера,

Bernoulli – числа и полиномы Бернулли,
 BesselI (J,K,Y), BesselI (J,K,Y) Zeros – функции Бесселя,
 Beta – бета функция
 Binomial – биномиальные коэффициенты,
 ChebyshevT(U) – функции Чебышева,
 Chi – интегральный гиперболический косинус,
 Ci – интегральный косинус
 CylinderD(U,V) – параболические функции,
 csgn – комплексная сигнум-функция,
 dilog – дилогарифм,
 Dirac – дельта-функция Дирака,
 Ei – экспоненциальный интеграл,
 EllipticCE(CK, CPi, E, F, K, Modulus, Nome, Pi) – эллиптические интегралы,
 erf – функция ошибок,
 erfc – дополнительная функция ошибок,
 euler – числа и полиномы Эйлера,
 FresnelC (f,g,S) – интегралы Френеля,
 GAMMA – гамма-функция,
 GaussAGM – арифметико-геометрическое среднее Гаусса,
 HankelH1(P2) – функции Ханкеля,
 Harmonic – частичные суммы гармонических рядов,
 Heaviside – функция Хевисайда,
 JacobiAM(P, CN, CD, DN, DC, DS, ND, NC, ND, SC, SD, SN) – эллиптические функции Якоби,
 InverseJacobiAM(CN, CD, DN, DC, DS, ND, NC, ND, SC, SD, SN) – функции, обратные к эллиптическим функциям Якоби,
 JacobiTheta1(2, 3, 4) – Тэта-функции Якоби,
 JacobiZeta – дзета-функция Якоби,
 KelvinBei(Ber, Hei, Her, Kei, Ker) – функции Кельвина,
 LambertW – W-функция Ламберта,
 Li – логарифмический интеграл,
 lnGAMMA – логарифмическая гамма-функция,
 MeijerG – G-функция Мейера,
 Pochhammer – символ Похгаммера,
 Polylog – полилогарифм,
 Psi – дигамма-функция,
 Shi – интегральный гиперболический синус,
 Si – интегральный синус,
 StruveH(L) – функции Струве,
 Surd – неглавная корневая функция,
 WeberE – E-функция Вебера,
 WeierstrassP – P-функция Вейерштрасса,
 WeierstrassPPrime – производная P-функции Вейерштрасса,
 WeierstrassSigma – сигма-функция Вейерштрасса,
 WeierstrassZeta – дзета- функция Вейерштрасса,
 WhittakerM (W) – функции Уиттекера,
 Zeta - дзета- функция Римана.

Примеры.

```

> WeierstrassP(1.0,2.0,3.0);
1.214433709

> WeierstrassPPrime(1.0,2.0,3.0);
-1.317406195
  
```

```

> WeierstrassZeta(1.0,2.0,3.0);
0.9443449465

> WeierstrassSigma(1.0,2.0,3.0);
0.9880674334

> JacobiNS(z,k) = 1/JacobiSN(z,k):
JacobiNC(z,k) = 1/JacobiCN(z,k):
JacobiND(z,k) = 1/JacobiDN(z,k):
> JacobiCS(z,k) = JacobiCN(z,k)/JacobiSN(z,k):
JacobiSC(z,k) = 1/JacobiCS(z,k):
> JacobiDS(z,k) = JacobiDN(z,k)/JacobiSN(z,k):
JacobiSD(z,k) = 1/JacobiDS(z,k):
> JacobiCD(z,k) = JacobiCN(z,k)/JacobiDN(z,k):
JacobiDC(z,k) = 1/JacobiCD(z,k):
> JacobiSN(z,k) = 'sin'(JacobiAM(z,k));
JacobiSN(z,k) = sin(JacobiAM(z,k))

> JacobiCN(z,k) = 'cos'(JacobiAM(z,k));
JacobiCN(z,k) = cos(JacobiAM(z,k))

> JacobiDN(z,k) = Diff(JacobiAM(z,k),z);
JacobiDN(z,k) =  $\frac{\partial}{\partial z}$  JacobiAM(z,k)

> JacobiDN(z,k)^2 = 1-k^2*JacobiSN(z,k)^2;
JacobiDN(z,k)^2 = 1 - k^2 JacobiSN(z,k)^2

> JacobiDN(k*z,(1/(k^2))^(1/2)) = JacobiCN(z,k);
JacobiDN $\left(kz, \sqrt{\frac{1}{k^2}}\right)$  = JacobiCN(z,k)

> F_trig := FunctionAdvisor( definition,
InverseJacobiAM(phi,k))[1];

```

$$F_{trig} := \text{InverseJacobiAM}(\phi, k) = \int_0^\phi \frac{1}{\sqrt{1 - k^2 \sin(\theta)^2}} d\theta$$

```

> FunctionAdvisor( definition, JacobiAM(phi,k));

$$\left[ \phi = \text{JacobiAM}\left(\int_0^\phi \frac{1}{\sqrt{1 - k^2 \sin(\theta)^2}} d\theta, k\right), \phi::\text{RealRange}\left(\frac{-3}{2}, \frac{3}{2}\right) \right]$$


```

```

> EllipticF(z,k):
% = convert(%, Int);

```

$$\text{EllipticF}(z, k) = \int_0^z \frac{1}{\sqrt{1 - \alpha^2} \sqrt{1 - k^2 \alpha^2}} d\alpha$$

```

> JacobiSN(z,k)^2 + JacobiCN(z,k)^2;
JacobiSN(z,k)^2 + JacobiCN(z,k)^2

> simplify(%);

```

```

1
> JacobiSN( InverseJacobiAM(z,k), k);
sin(z)
> JacobiCN( InverseJacobiAM(z,k), k);
cos(z)
> JacobiDN( InverseJacobiAM(z,k), k);
 $\sqrt{1 - k^2 \sin(z)^2}$ 
> JacobiSN( InverseJacobiCN(z,k), k);
 $\sqrt{1 - z^2}$ 
> JacobiCN( InverseJacobiSN(z,k), k);
 $\sqrt{1 - z^2}$ 
> JacobiTheta1(2.0,0.5);
1.632025908
> JacobiTheta2(2.0,0.5);
-0.3181628211
> JacobiTheta3(2.0,0.5);
0.3314359778
> JacobiTheta4(2.0,0.5);
1.632130567
> Ci(1.);
0.3374039229
> Ci(3);
Ci(3)
> evalf(%);
0.1196297860
> Si(3.14159+7.6*I);
63.60695388- 123.1816272I
> Ssi(12345.67890);
-0.00005756635677
> si(12345.67890);
1.570738760
> Shi(Pi);
Shi( $\pi$ )
> Chi(1.+I);
0.8821721806+ 1.283547193I
> convert(Ci(x), Ei);
 $-\frac{1}{2} \text{Ei}(1, -I x) - \frac{1}{2} \text{Ei}(1, x I) + \frac{1}{2} (\pi \text{csgn}(x) I - \pi I) \text{csgn}(x I)$ 
> BesselJ(0,2);
BesselJ(0, 2)

```

```

> evalf(%);
                                0.2238907791

> BesselK(1,-3.);
                                -0.04015643113- 12.41987883I

> BesselI(0,0);
                                1

> BesselY(1.5+I,3.5-I);
                                0.9566518512- 1.465483431I

> series(BesselJ(3,x),x);
                                 $\frac{1}{48}x^3 - \frac{1}{768}x^5 + O(x^6)$ 

> diff(BesselJ(v,x),x);
                                 $-\text{BesselJ}(v+1, x) + \frac{v \text{BesselJ}(v, x)}{x}$ 

> HankelH1(2.5,3.7+I);
                                0.1809260572- 0.08706107529I

> diff(HankelH2(v,x^2),x);
                                 $2 \left( -\text{HankelH2}(v+1, x^2) + \frac{v \text{HankelH2}(v, x^2)}{x^2} \right) x$ 

> convert(HankelH2(v,x),Bessel);
                                 $\text{BesselJ}(v, x) - \text{BesselY}(v, x) I$ 

> convert(AiryAi(x),Bessel);
                                 $\frac{1}{3} \frac{-x \text{BesselI}\left(\frac{1}{3}, \frac{2}{3} \sqrt{x^3}\right) + \text{BesselI}\left(\frac{-1}{3}, \frac{2}{3} \sqrt{x^3}\right) (x^3)^{(1/3)}}{(x^3)^{(1/6)}}$ 

> convert(KelvinKer(v,x),BesselK);
                                 $\frac{1}{2} \frac{\text{BesselK}\left(v, \left(\frac{1}{2} + \frac{1}{2} I\right) x \sqrt{2}\right) + (e^{(1/2 I v \pi)})^2 \text{BesselK}\left(v, \left(\frac{1}{2} - \frac{1}{2} I\right) x \sqrt{2}\right)}{e^{(1/2 I v \pi)}}$ 

> 'Dirac(-x)' = Dirac(-x);
                                Dirac(-x) = Dirac(x)

> Dirac(c*x) = 1/abs(c)*Dirac(x);
                                 $\text{Dirac}(c x) = \frac{\text{Dirac}(x)}{|c|}$ 

> harmonic(3);
                                 $\frac{11}{6}$ 

> harmonic(3,2);
                                 $\frac{49}{36}$ 

> harmonic(r,s):
% = convert(%, Sum) assuming r::nonnegint;

```

$$\text{harmonic}(r, s) = \sum_{k=1}^r \frac{1}{k^s}$$

```
> %% = convert(%%, Zeta);
harmonic(r, s) = ζ(s) - ζ(0, s, r + 1)

> %%% = convert(%%%, Psi) assuming s::posint;
harmonic(r, s) = 
$$\frac{(-1)^s (\Psi(-1 + s, 1) - \Psi(-1 + s, r + 1))}{(-1 + s)!}$$


> diff(%, r);
s ζ(0, s + 1, r + 1) = - 
$$\frac{(-1)^s \Psi(s, r + 1)}{(-1 + s)!}$$


> evalf( eval(%, [r = 10/43 + I/2, s = 4]) );
-0.2942981267- 0.9671639794I = -0.2942981267- 0.9671639794I
```

Справку о всех имеющихся в Maple функциях можно получить, выполнив команду **?inifunction**, или через меню **Help/Topic Search/inifunction**.

Элементы программирования в Maple.

Простейшим способом задания функции пользователя является применение функционального оператора. При этом используется следующая конструкция.

> **name := (x, y, ...,) -> expr**. После этого вызов функции осуществляется в виде **name := (x, y, ...,) -> expr**, где (x, y, ...) список формальных параметров функции пользователя с именем **name**. Переменные, указанные в списке параметров, являются локальными. При подстановке на их место фактических параметров они сохраняют их значения только в теле функции **expr**. За пределами этой функции переменные с этими именами оказываются либо неопределенными, либо сохраняют ранее присвоенные им значения.

Иногда удобно перед группой действий вставлять команду **restart**. Эта команда приводит все ячейки и имена в исходное состояние.

Пример.

```
> restart;
> x:=0; y:=0;
x := 0
y := 0

> m:=(x,y)->sqrt(x^2+y^2);
m := (x, y) →  $\sqrt{x^2 + y^2}$ 

> m(3,4);
5
```

Для написания разветвляющихся программ в Maple имеется условный оператор **if**, позволяющий создавать следующие конструкции

```
if <условие> then <действия>
| elif <условие> then <действия>
| else <действия>
end if;
```

В прямых скобках находятся необязательный элемент конструкции. Чаще всего используются две следующих конструкции.

if <условие А> then <действия 1> end if – если условие А выполняется, то выполняются действия 1, в противном случае ничего не выполняется;

if<условие A >**then**< действия 1> **else**< действия 2> **end if** - если условие A выполняется, то выполняются действия 1, в противном случае выполняются действия 2.

В задании условий используются любые логические конструкции со знаками сравнения (<, <=, >, >=, <>) и логические операторы **and**, **or** и **not**, конструкции с которыми возвращают значения **true** и **false**.

Замечание. В ранних версиях Maple признаком конца условной операции являлось слово **fi**.

Пример.

```
> a := 3; b := 5;
                                a := 3
                                b := 5

> if (a > b) then a else b end if;
                                5

> 5*(Pi + `if`(a > b,a,b));
                                5 π + 25

> x := `if`(a < b,NULL,b);
                                x :=

> if FAIL then 3 else 5 end if;
                                5
```

Циклы for и while.

Часто требуется циклические повторения выражения заданное число раз или до тех пор, пока выполняется заданное условие. В Maple имеется общая конструкция цикла, которая задается следующим образом.

```
| for <name> || from <expr1> || by <expr3> > || to <expr2> || while <expr4> |
do <statement sequence> end do;
```

Здесь **name** - имя управляющей переменной цикла (счетчик), **expr1**, **expr2**, и **expr3** - выражения задающие начальное значение, конечное значение и шаг изменения переменной **name**, **expr4** - выражение, задающее условие, пока цикл (**statement sequence** - действия между словами **do** и **od**) будет выполняться.

В ходе выполнения цикла управляющая переменная меняется от начального значения **expr1** до значения **expr2** с шагом, заданным **expr3**. Если блок **by <expr3>** отсутствует, то управляющая переменная будет меняться с шагом, равным +1 при **expr1 < expr2**.

Примеры.

```
1) > for i from 1 to 5 do print (i) od;
                                1
                                2
                                3
                                4
                                5

2) > tot := 0;
for i from 11 by 2 while i < 100 do
    tot := tot + i
end do;

3) > tot:=0;
for z in bob do
```

```

    tot:=tot+z
end do;
4)> tot := 1;
for z in 1, x, y, q^2, 3 do
    tot := tot*z;
end do;

```

Для того, чтобы пропустить заданное значение управляющей переменной или остановить выполнение цикла после заданного значения управляющей переменной используются оператор пропуска (**next**) и оператор прерывания (**break**). Заметим, что вместе с признаком конца цикла **od** используется его эквивалент - слово **end do**.

Примеры.

```

> for n from 1 to 3 do
if n=2 then next; end if;
print(n);
end do;

```

1

3

```

> L := [1, 2, "abc", "a", 7.0, infinity]:
for x in L do
if type(x, 'string') then
print(x);
break;
end if;
end do;

```

"abc"

Процедуры (подпрограммы).

Процедурой называют модуль программы, имеющий самостоятельное значение и выполняющий, обычно, действия достаточно сложные и отличные от действий, выполняемых в программе.

Простейшая форма задания процедуры следующая.

Name:=proc(параметры)

Тело процедуры

end

Параметры процедуры задаются перечислением имен переменных, например, **proc**(x,y,z). С помощью оператора **::** после имени переменной можно определить ее тип. Например, в объявлении **proc**(n:: integer) объявляется, что переменная **proc**(x,y,z) является целочисленной.

Пример.

```

> lc := proc( s, u, t, v )
    description "form a linear combination of the arguments";
    s * u + t * v
end proc;

```

lc := proc(s, u, t, v)

description "form a linear combination of the arguments";

s×u + t×v

end proc

```

> print( lc );

```

```

proc(s, u, t, v)
description "form a linear combination of the arguments";
      s×u + t×v
end proc

```

```

> lc( Pi, x, -I, y );

```

$$\pi x - y I$$

Вычисления и символьные операции в системе Maple

Вычисления и операции с формулами.

Вычисления сумм последовательностей.

Для вычисления сумм последовательности используется формула

$$\sum_{k=m}^n f(k) = f(m) + \dots + f(n).$$

Для вызова этой операции используются следующие команды:

Sum (f,k), sum (f,k), Sum (f,k=m..n), sum (f,k=m..n),

Примеры.

```

> restart;
> sum(k^2,k=1..10);

```

$$385$$

```

> A:=sum (1/k^2,k=1..100);
A := 15895086941330378731122979285175538597023834985437098598894328348(
      8131090369901 / 9721861444343810305896579766726231441619755839957(
      41782720354705517986165248000

```

```

> evalf(A,100);
1.63498390018489286507716949818032376668332170003126381385311098048563(
      881260641124619411444400948

```

```

> B:=sum_{k=1}^{\infty} 1/k^2

```

$$B := \frac{\pi^2}{6}$$

```

> B:=sum (1/k^2,k=1..infinity);

```

$$B := \frac{\pi^2}{6}$$

```

> evalf(B,100);
1.64493406684822643647241516664602518921894990120679843773555822937000(
      403200873833628900619758706

```

```

> sum(k^(3/2), k=1..infinity);

```

$$\infty$$

```

> S:=sum(k^2,k=1..2007);

```

$$S := 2696779140$$

Вычисление производных.

Для вычисления производных (обычных и частных) используются команды D, d, diff и Diff.

Команда diff имеет следующий вид:

`diff(f, x1, ..., xj), diff(f, [x1$n]), diff(f, x1$n, [x2$n, x3], ... xi, [xj$m]).`

В инертной форме используется команда Diff.

Примеры.

`> diff(sin(x), x);`

$$\cos(x)$$

`> diff(sin(x), y);`

$$0$$

`> diff(sin(x), x$3);`

$$-\cos(x)$$

`> diff(x*sin(cos(x)), x);`

$$\sin(\cos(x)) - x \cos(\cos(x)) \sin(x)$$

`> diff(tan(x), x);`

$$1 + \tan(x)^2$$

`> Diff(tan(x), x);`

$$\frac{d}{dx} \tan(x)$$

`> Diff(tan(x), x) = diff(tan(x), x);`

$$\frac{d}{dx} \tan(x) = 1 + \tan(x)^2$$

`> diff(f(x), x);`

$$\frac{d}{dx} f(x)$$

`> diff(f(x,y), x,y);`

$$\frac{\partial^2}{\partial y \partial x} f(x, y)$$

`> diff(f(x,y), x,y) - diff(f(x,y), y,x);`

$$0$$

`> diff(g(x,y,z), x,z,z);`

$$\frac{\partial^3}{\partial z^2 \partial x} g(x, y, z)$$

`> diff(g(x,y,z), [x,z,z]);`

$$\frac{\partial^3}{\partial z^2 \partial x} g(x, y, z)$$

`> Diff(sin(x), x$n);`

$$\frac{d^n}{dx^n} \sin(x)$$

`> value(%);`

$$\sin\left(x + \frac{1}{2} n \pi\right)$$

`> Diff(exp(x^2), x$n);`

$$\frac{d^n}{dx^n} (e^{(x^2)})$$

> value(%);

$$x^{(-n)} 2^n \text{MeijerG}\left(\left[\left[0, \frac{1}{2}\right], [1]\right], \left[[0], \left[\frac{1}{2} + \frac{1}{2}n, \frac{1}{2}n\right]\right], -x^2\right)$$

> Diff(f(x)*g(x), x\$n);

$$\frac{d^n}{dx^n} (f(x) g(x))$$

> value(%);

$$\sum_{k=0}^n \text{binomial}(n, k) \left(\frac{d^k}{dx^k} f(x) \right) \left(\frac{d^{n-k}}{dx^{n-k}} g(x) \right)$$

Для дифференцирования используют также оператор **D**. Этот оператор можно записывать в следующих формах: **D(f)**, **D[i](f)**, **D[i](f)(x, y, ...)**.

Примеры.

> D[1,1,2](g)(x,y);

$$D_{1,1,2}(g)(x, y)$$

> convert(%, diff);

$$\frac{\partial^3}{\partial y \partial x^2} g(x, y)$$

> D(f)(0);

$$D(f)(0)$$

> convert(%, diff);

$$\left(\frac{d}{dt} f(t) \right) \Big|_{t=0}$$

> D[2](F)(t,v(t));

$$D_2(F)(t, v(t))$$

> convert(%, diff);

$$\left(\frac{\partial}{\partial t} F(t, t) \right) \Big|_{t=v(t)}$$

> D(f) = (q -> eval(diff(f(t),t), t=q));

$$D(f) = \left(q \rightarrow \left(\frac{d}{dt} f(t) \right) \Big|_{t=q} \right)$$

> %(x);

$$D(f)(x) = \frac{d}{dx} f(x)$$

> %(x+1);

$$D(f)(x+1) = \left(\frac{d}{dt} f(t) \right) \Big|_{t=x+1}$$

> D(y)(x);

$$D(y)(x)$$

> convert(%, diff);

```


$$\frac{d}{dx} y(x)$$

> convert(%, D);

$$D(y)(x)$$

> D(sin);

$$\cos$$

> (D@@n)(sin); # nth derivative mapping

$$z \rightarrow \sin\left(z + \frac{1}{2} n \pi\right)$$

> D[1$n](sin); # alternative input

$$z \rightarrow \sin\left(z + \frac{1}{2} n \pi\right)$$

> D(f); # f represents an arbitrary unknown mapping

$$D(f)$$

> D(D(f)); # second derivative mapping

$$(D^{(2)})(f)$$

> (D@@2)(f); # alternative input

$$(D^{(2)})(f)$$

> D[1$n,2$m](g); # partial derivative mapping of order n + m

$$D_{1\ \$n, 2\ \$m}(g)$$

> D[i,j](g) - D[j,i](g); # partial derivatives are assumed to commute

$$0$$

> D[1](D[2,1](g)); # derivative mapping of a derivative mapping

$$D_{1,1,2}(g)$$

> D(arctan);

$$z \rightarrow \frac{1}{1+z^2}$$


```

Вычисление определенных и неопределенных интегралов.

Для вычисления неопределенных интегралов используются функции `int(f,x)` и `Int(f,x)` (инертная форма). Для вычисления определенных интегралов используются функции `int(f,x=a..b)` и `Int(f,x)` (инертная форма).

Примеры.

```

> int( sin(x), x );

$$-\cos(x)$$

> int( sin(x), x=0..Pi );

$$2$$

> int( sin, a..b );

$$\cos(a) - \cos(b)$$

> int( x/(x^3-1), x );

```

$$\frac{1}{3} \ln(x-1) - \frac{1}{6} \ln(x^2+x+1) + \frac{1}{3} \sqrt{3} \arctan\left(\frac{1}{3}(2x+1)\sqrt{3}\right)$$

> `int(exp(-x^2), x);`

$$\frac{1}{2} \sqrt{\pi} \operatorname{erf}(x)$$

> `int(exp(-x^2)*ln(x), x);`

$$\int e^{(-x^2)} \ln(x) dx$$

> `int(exp(-x^2)*ln(x), x=0..infinity);`

$$-\frac{1}{4} \sqrt{\pi} \gamma - \frac{1}{2} \sqrt{\pi} \ln(2)$$

> `int(exp(-x^2)*ln(x)^2, x=0..infinity);`

$$\frac{1}{16} \pi^{(5/2)} + \frac{1}{8} \sqrt{\pi} \gamma^2 + \frac{1}{2} \sqrt{\pi} \gamma \ln(2) + \frac{1}{2} \sqrt{\pi} \ln(2)^2$$

$$> A := \int_{-\infty}^{\infty} \frac{\sin(x)}{x} dx$$

;

$$A := \pi$$

> `evalf(A,100);`

$$3.14159265358979323846264338327950288419716939937510582097494459230781\backslash$$

$$286208998628034825342117068$$

Вычисление пределов функций.

Для вычисления пределов функции f в точке $x=a$ используются следующие функции: **limit(f, x=a)**, **limit(f, x=a, dir)**, где f – алгебраическое выражение, x – имя переменной, dir – направление поиска предела (left – слева, right – справа).

Примеры.

> `limit(sin(x)/x, x=0);`

$$1$$

> `limit(exp(x), x=infinity);`

$$\infty$$

> `limit(exp(x), x=-infinity);`

$$0$$

> `limit(1/x, x=0, real);`

undefined

> `limit(exp(x^2)*(1-erf(x)), x=infinity);`

$$0$$

Разложение функций в ряды.

Для разложения функции в степенной ряд используются следующие функции: **series(expr, eqn)** и **series(expr, eqn, n)**, где **expr** – разлагаемое выражение, **eqn** – условие (например, в виде $x=a$) или имя переменной (например, x) и **n** – необязательный параметр, задающий число членов ряда (при его отсутствии оно равно 6 и может переустанавливаться системой переменной **order**)/

Примеры.

```

> series(x/(1-x-x^2), x=0);

$$x + x^2 + 2x^3 + 3x^4 + 5x^5 + O(x^6)$$

> convert(%,polynom);

$$x + x^2 + 2x^3 + 3x^4 + 5x^5$$

> series(x+1/x, x=1, 3 );

$$2 + (x-1)^2 + O((x-1)^3)$$

> series(exp(x), x=0, 8 );

$$1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + \frac{1}{24}x^4 + \frac{1}{120}x^5 + \frac{1}{720}x^6 + \frac{1}{5040}x^7 + O(x^8)$$

> series(exp(x)/x, x=0, 8 );

$$x^{-1} + 1 + \frac{1}{2}x + \frac{1}{6}x^2 + \frac{1}{24}x^3 + \frac{1}{120}x^4 + \frac{1}{720}x^5 + \frac{1}{5040}x^6 + O(x^7)$$

> series(GAMMA(x), x=0, 2 );

$$x^{-1} - \gamma + \left( \frac{1}{12}\pi^2 + \frac{1}{2}\gamma^2 \right)x + O(x^2)$$

> series(x^3/(x^4+4*x-5),x=infinity);

$$\frac{1}{x} - \frac{4}{x^4} + \frac{5}{x^5} + O\left(\frac{1}{x^7}\right)$$

> int(exp(x^3), x );

$$-\frac{1}{3}(-1)^{(2/3)} \left( \frac{2}{3} \frac{x(-1)^{(1/3)} \pi \sqrt{3}}{\Gamma\left(\frac{2}{3}\right)(-x^3)^{(1/3)}} - \frac{x(-1)^{(1/3)} \Gamma\left(\frac{1}{3}, -x^3\right)}{(-x^3)^{(1/3)}} \right)$$

> series(%, x=0);

$$x + \frac{1}{4}x^4 + O(x^7)$$

> series(x^x, x=0, 3);

$$1 + \ln(x)x + \frac{1}{2}\ln(x)^2 x^2 + O(x^3)$$


```

Решение уравнений и неравенств.

Решение одиночных уравнений вида $f(x)=0$ обеспечивается функцией **solve(f(x),x)**.

Примеры.

```

> solve( 2*y - (x - 1)^2 = 2, y );

$$\frac{1}{2}x^2 - x + \frac{3}{2}$$

> solve( x^2 - x = 2025, x );

$$\frac{1}{2} + \frac{1}{2}\sqrt{8101}, \frac{1}{2} - \frac{1}{2}\sqrt{8101}$$

> solve( x^4 - x^3 = -1, x );
RootOf(_Z^4 - _Z^3 + 1, index = 1), RootOf(_Z^4 - _Z^3 + 1, index = 2),
RootOf(_Z^4 - _Z^3 + 1, index = 3), RootOf(_Z^4 - _Z^3 + 1, index = 4)
> evalf(%);

```

$$1.018912794 + 0.6025654200i, -0.5189127944 + 0.6666098449i, \\ -0.5189127944 - 0.6666098449i, 1.018912794 - 0.6025654200i$$

Решение тригонометрических уравнений.

Функция может использоваться для решения тригонометрических уравнений:

```
> solve( sin(x) -1/2, x );
```

$$\frac{\pi}{6}$$

Однако здесь предъявлено лишь одно решение уравнения. Для получения всех решений используется конструкция **_EnvAllSolutions**:

```
> _EnvAllSolutions := true:
```

```
solve( sin(x) -1/2, x );
```

$$\frac{1}{6}\pi + \frac{2}{3}\pi_B2\sim + 2\pi_Z4\sim$$

Здесь встречаются переменные **_B2~** и **_Z4~**, означающие, соответственно, бинарное переменное (принимающее значение 0 и 1) и целое переменное (принимающее целые значения).

Решение систем линейных и нелинейных уравнений.

Для решения систем линейных уравнений существуют мощные матричные методы, которые будут рассмотрены позже. Однако функция **solve** также с успехом справляется с этой задачей. Формат команды **solve** для решения систем уравнений имеет вид:

```
solve({eq1,...,eqn}, [x1,...,xn])
```

Примеры.

```
> solve( {3.2*x + 1.3*y + 4.2*z = 5, 8.7*x + 19*y + 11.2*z = 94, x + y/4 + z = 1}, [x, y, z]);
```

$$[[x = 0.4969502408, y = 5.187800963, z = -0.7939004815]]$$

```
> solve( {x + y = 10, x^2 = 9}, [x, y] );
```

$$[[x = 3, y = 7], [x = -3, y = 13]]$$

```
> RootOf(x^2+1=0,x);
```

$$\text{RootOf}(_Z^2 + 1)$$

```
> allvalues(%);
```

$$I, -I$$

Решение неравенств.

Для решения неравенств, систем неравенств, систем равенств и неравенств используется та же команда с заменой в соответствующем месте равенств на неравенства.

Примеры.

```
> solve( x^2 - 5*x + 6 > 0, x );
```

$$\text{RealRange}(-\infty, \text{Open}(2)), \text{RealRange}(\text{Open}(3), \infty)$$

```
> solve( x^2 - 5*x + 6 <= 0, x );
```

$$\text{RealRange}(2, 3)$$

```
> solve( {x + y < 10, x^2 = 9}, [x, y] );
```

$$[[y < 13, x = -3], [x = 3, y < 7]]$$

Анализ функций.

Отыскание минимумов и максимумов функций, заданных аналитически.

Для отыскания экстремумов заданной функции используются функции: **extrema(expr, constraints)**, **extrema(expr, constraints, vars)**, **extrema(expr, constraints, vars, 's')**, где **expr** - исследуемая функция, **constraints** - ограничения, при которых ищется экстремум (условный экстремум), **vars** - переменные и **'s'** - переменная, которой присваиваются координаты найденной точки экстремума.

Чтобы найти минимум или максимум функции используют функции:

minimize(expr.opt1,...,optn), **maximize(expr.opt1,...,optn)**.

Эти функции могут разыскивать максимумы и минимумы для функций одного и нескольких переменных. С помощью опций **opt1,...,optn** можно указывать дополнительные данные для поиска. Например, параметр **infinity** означает, что поиск выполняется на всей числовой прямой, а параметр **location** (или **location=true**) дает расширенный вывод результата: не только значение экстремума, но и значения переменных в этой точке.

Примеры.

```
>
extrema( a*x^2+b*x+c, {}, x );

$$\left\{ \frac{1}{4} \frac{-b^2 + 4ca}{a} \right\}$$


> extrema( a*x*y*z, x^2+y^2+z^2=1, {x,y,z} );

$$\left\{ \min\left(\frac{1}{3} \% 1, -\frac{1}{3} \% 1, 0\right), \max\left(\frac{1}{3} \% 1, -\frac{1}{3} \% 1, 0\right) \right\}$$


%1 := a RootOf(3 _Z^2 - 1)

> f := (x^2+y^2)^(1/2)-z; g1 := x^2+y^2-16; g2 := x+y+z = 10;

$$f := \sqrt{x^2 + y^2} - z$$


$$g1 := x^2 + y^2 - 16$$


$$g2 := x + y + z = 10$$


> extrema(f, {g1,g2}, {x,y,z}, 's');
{ min(4 RootOf(_Z^2 - 2, label=_L5) - 6), max(4 RootOf(_Z^2 - 2, label=_L5) - 6) }

> s;
{ { y = 2 RootOf(_Z^2 - 2, label=_L5), x = 2 RootOf(_Z^2 - 2, label=_L5),
  z = -4 RootOf(_Z^2 - 2, label=_L5) + 10 } }

> minimize((x-1)^2+2*(y-2)^2+1, location);
1, { [{ y = 2, x = 1 }, 1] }
```

Анализ функции на непрерывность.

Для исследования функции на непрерывность используется функция **iscont**.

Примеры.

```
> iscont( 1/x, x=1..2 );
true

> iscont( 1/x, x=-1..1 );
false

> iscont( 1/x, x=0..1 );
true

> iscont( 1/x, x=0..1, 'closed' );
false
```

Для создания кусочно-заданной функции используется функция **piecewise**.

Примеры.

> **piecewise(x>0,x);**

$$\begin{cases} x & 0 < x \\ 0 & \text{otherwise} \end{cases}$$

> **piecewise(x*x>4 and x<8,f(x));**

$$\begin{cases} f(x) & 4 < x^2 \text{ and } x < 8 \\ 0 & \text{otherwise} \end{cases}$$

> **simplify(%);**

$$\begin{cases} f(x) & x < -2 \\ 0 & x \leq 2 \\ f(x) & x < 8 \\ 0 & 8 \leq x \end{cases}$$

> **assume(a<b,b<c);**

piecewise(x>a and x<b,1,x>b and x<c,2);

$$\begin{cases} 1 & a \sim < x \text{ and } x < b \sim \\ 2 & b \sim < x \text{ and } x < c \sim \end{cases}$$

> **convert(%,piecewise,x);**

$$\begin{cases} 0 & x \leq a \sim \\ 1 & x < b \sim \\ 0 & x = b \sim \\ 2 & x < c \sim \\ 0 & c \sim \leq x \end{cases}$$

> **f:=x->piecewise(x<0,-1,x<1,0,1);**

$$f := x \rightarrow \text{piecewise}(x < 0, -1, x < 1, 0, 1)$$

> **f(1/2);**

$$0$$

> **p := piecewise(x<0,-x,x>0,x);**

$$p := \begin{cases} -x & x < 0 \\ x & 0 < x \end{cases}$$

> **p^2 + 5;**

$$\left(\begin{cases} -x & x < 0 \\ x & 0 < x \end{cases} \right)^2 + 5$$

> **simplify(p*p);**

$$\left(\begin{cases} x & x < a \sim \\ x^2 & b \sim < x \end{cases} \right)^2$$

> **convert(%,piecewise,x);**

$$\begin{cases} x^2 & x < a \sim \\ 0 & x \leq b \sim \\ x^4 & b \sim < x \end{cases}$$

> **assume(a<0);**

piecewise(x*a-2>0,1,2);

$$\begin{cases} 1 & 0 < x \leq -2 \\ 2 & \text{otherwise} \end{cases}$$

> **convert(% ,piecewise,x);**

$$\begin{cases} 1 & x < \frac{2}{a} \\ 2 & \frac{2}{a} \leq x \end{cases}$$

Некоторые действия с полиномами.

Для выделения коэффициентов полиномов используются следующие функции:

coeff(p, x), coeff(p, x, n), coeff(p, x^n), collect(p, x)

Здесь функция **coeff(p, x)** возвращает коэффициент при x в полиноме **p**, **coeff(p, x, n)** – возвращает коэффициент для члена со степенью **n**, **coeff(p, x^n)** – возвращает коэффициенты при x^n , **collect(p, x)** – возвращает полином, объединяя коэффициенты при степенях x .

Примеры.

> **p := 2*x^2 + 3*y^3 - 5;**

coeff(p,x,2);

2

> **coeff(p,x^2);**

2

> **coeff(p,x,0);**

$3y^3 - 5$

> **q := 3*a*(x+1)^2 + sin(a)*x^2*y - y^2*x + x - a;**

coeff(q,x);

$6a - y^2 + 1$

Для разложения полинома (а также целого числа) на множители используется функция **factor**.

Примеры.

> **factor(6*x^2+18*x-24);**

$6(x+4)(x-1)$

> **factor(6);**

6

> **ifactor(6);**

(2) (3)

> **factor((x^3-y^3)/(x^4-y^4));**

$\frac{y^2 + yx + x^2}{(x+y)(y^2 + x^2)}$

> **factor(x^3+5);**

$x^3 + 5$

> **factor(x^3+5, 5^(1/3));**

$(x^2 - x5^{(1/3)} + 5^{(2/3)})(x + 5^{(1/3)})$

> **factor(x^3+5, {5^(1/3), (-3)^(1/2)});**

$\frac{1}{4}(2x - 5^{(1/3)} - \sqrt{-3}5^{(1/3)})(2x - 5^{(1/3)} + \sqrt{-3}5^{(1/3)})(x + 5^{(1/3)})$

```
> factor(x^3+5.0);
(x + 1.709975947) (x^2 - 1.709975947x + 2.924017740)

> factor(x^3+5,complex);
(x + 1.709975947) (x - 0.8549879733+ 1.480882610I)
(x - 0.8549879733- 1.480882610I)

> factor(y^4-2,sqrt(2));
(y^2 + sqrt(2)) (y^2 - sqrt(2))
```

Для вычисления корней полинома используются уже отмеченная функция **solve** и функция **roots**.

Примеры.

```
> roots(2*x^3+11*x^2+12*x-9);
[[[1/2, 1], [-3, 2]]]

> roots(x^4-4);
[]

> roots(x^4-4,x);
[]

> solve(x^4-4,x);
sqrt(2) I, -I sqrt(2), sqrt(2), -sqrt(2)

> roots(x^3+(-6-b-a)*x^2+(6*a+5+5*b+a*b)*x-5*a-5*a*b,x);
[[5, 1]]

> roots(x^4-4, sqrt(2));
[[sqrt(2), 1], [-sqrt(2), 1]]

> roots(x^4-4, {sqrt(2),I});
[[sqrt(2), 1], [-sqrt(2), 1], [-I sqrt(2), 1], [sqrt(2) I, 1]]
```

Символьные операции.

Одним из видов символьных преобразований является преобразование выражения в тождественную форму. Основной функцией для такого преобразования является функция **convert(expr, form, arg)**, где - выражение, - название формы, - необязательный аргумент.

Примеры.

```
> convert(sin(I*x),exp);
(1/2 e^x - 1/2 1/e^x) I

> convert(arcsinh(x),ln);
ln(x + sqrt(x^2 + 1))

> convert(123,binary);
1111011

> s:=taylor (sin(x),x,8);
s := x - 1/6 x^3 + 1/120 x^5 - 1/5040 x^7 + O(x^8)
```

> **p:=convert(s,polynomial);**

$$p := x - \frac{1}{6}x^3 + \frac{1}{120}x^5 - \frac{1}{5040}x^7$$

> **q:=convert(p,float);**

$$q := x - 0.1666666667x^3 + 0.008333333333x^5 - 0.0001984126984x^7$$

Другим средством преобразования выражений является функция **combine**. Она обеспечивает объединение показателей степенных функций и преобразование тригонометрических и некоторых других функций. Эта функция может иметь один из следующих видов:

combine (f), combine (f, n), combine (f, n, opt1, opt2,...).

Здесь **f** - выражение, множество или список выражений; **n** - имя, множество или список имен; **opt1, opt2,...** - имена параметров. Во втором аргументе можно использовать следующие функции: **@@, abs, arctan, conjugate, exp, ln, piecewise, polylog, power, product, ps, radical, range, signum, trig**.

Примеры.

> **combine(Int(x,x=a..b)-Int(x^2,x=a..b));**

$$\int_a^b -x^2 + x \, dx$$

> **combine(Limit(x,x=a)*Limit(x^2,x=a)+c);**

$$\lim_{x \rightarrow a} x^3 + c$$

> **combine(4*sin(x)^3,trig);**

$$-\sin(3x) + 3 \sin(x)$$

> **combine(exp(x)^2*exp(y),exp);**

$$e^{(2x+y)}$$

> **combine(exp(sin(a)*cos(b))*exp(cos(a)*sin(b)),[trig,exp]);**

$$e^{\sin(a+b)}$$

> **assume(y>0,z>0);**

combine(2*ln(y)-ln(z),ln);

$$\ln\left(\frac{y^2}{z}\right)$$

> **combine((x^a)^2,power);**

$$x^{(2a)}$$

> **combine(Psi(-x)+Psi(x),Psi);**

$$2\Psi(x) + \pi \cot(\pi x) + \frac{1}{x}$$

> **combine([2*sin(x)*cos(x),2*cos(x)^2-1],trig);**

$$[\sin(2x), \cos(2x)]$$

> **combine(exp(sin(a)*cos(b))*exp(cos(a)*sin(b)),[trig,exp]);**

$$e^{\sin(a+b)}$$

> **combine(4*sin(x)^3+exp(y)*exp(x));**

$$-\sin(3x) + 3 \sin(x) + e^{(y+x)}$$

> **combine(4*sin(x)^3+exp(y)*exp(x),trig);**

$$-\sin(3x) + 3 \sin(x) + e^{y \sim} e^x$$

```
> combine(4*sin(x)^3+exp(y)*exp(x),exp);
```

$$4 \sin(x)^3 + e^{(y \sim + x)}$$

К символьным преобразованиям относится преобразование подстановки – замена подвыражения в заданном выражении на функцию от этого подвыражения. Для этого можно воспользоваться функцией **applyop**. Она используется в двух формах:

Applyop(f, i, e) – применяет функцию f к i – му подвыражению выражения e;

Applyop(f, i, e, ...xk, ...) – применяет функцию f к i – му подвыражению выражения e с передачей необязательных дополнительных аргументов xk;

Примеры.

```
> p := y^2-2*y-3;
```

$$p := y^2 - 2y - 3$$

```
> applyop(f,2,p);
```

$$y^2 + f(-2y) - 3$$

```
> applyop(f,2,p,x1,x2);
```

$$y^2 + f(-2y, x1, x2) - 3$$

```
> applyop(f,[2,2],p);
```

$$y^2 - 2f(y) - 3$$

```
> applyop(f,{2,3},p);
```

$$y^2 + f(-2y) + f(-3)$$

```
> applyop(abs,[2,1],3,p);
```

$$y^2 + 2y + 3$$

```
> e := (z+1)*ln(z*(z^2-2));
```

$$e := (z + 1) \ln(z(z^2 - 2))$$

```
> expand(e);
```

$$\ln(z(z^2 - 2))z + \ln(z(z^2 - 2))$$

К подстановкам относятся функциональные преобразования элементов списков. Эти преобразования можно производить с помощью следующих функций;

map(fcn, expr, arg1, ... , argn), **map**(fcn, arg1, ... , argn).

Здесь **fcn** - процедура или имя, **expr** - выражение, **argi** - необязательные дополнительные аргументы для **fcn**.

Примеры.

```
> map(f, x + y*z);
```

$$f(x) + f(yz)$$

```
> map(f, y*z);
```

$$f(y)f(z)$$

```
> map(f, {a,b,c});
```

$$\{f(a), f(b), f(c)\}$$

```
> map(x -> x^2, x + y);
```

$$x^2 + y^2$$

```
> map(proc(x,y) x^2+y end proc, [1,2,3,4], 2);
```

$$[3, 6, 11, 18]$$

```

> map2(f, g, {a,b,c});
                                { f(g, a), f(g, b), f(g, c) }

> map2(op, 1, [a+b,c+d,e+f]);
                                [a, c, e]

> sum_to_product:=proc(expr)
  if expr::atomic then
    expr;
  elif expr::`+` then
    `*`(op(expr));
  else
    map(procname,expr); end if;
end proc;
sum_to_product(3+4*x^2+3*(x+2)*(x+7)^2);
                                 $36x^2(x+2)(x+7)^2$ 

```

Популярными формами подстановок являются подстановки с помощью функций **subs** и **subsop**. Эти функции имеют следующий вид.

subs(x=a, e) – в выражении **e** заменяет подвыражение **x** на подвыражение **a**;

subs(s1, ..., sn, e) – в выражении **e** заменяет одни подвыражения на другие, выбирая их из списков **s1, ..., sn** вида **x=a**;

subsop(eq1, ..., eqn, e) – в выражении **e** заменяет указанные в **eqi** операнды другими, указанными в правой части равенств **eqi** вида **ni=ei**, где **ni** – номер операнда, **ei** – выражение для замены.

Примеры.

```

> subs( x=2, x^2+x+1 );
                                7

> subs( x=r^(1/3), 3*x*ln(x^3) );
                                 $3r^{(1/3)}\ln(r)$ 

> subs( sin(x)=y, sin(x)/sqrt(1-sin(x)) );
                                 $\frac{y}{\sqrt{1-y}}$ 

> subs( a+b=y, (a+b+c)^2 );
                                 $(a+b+c)^2$ 

> subs( a^2=y, a^3 );
                                 $a^3$ 

> algsups( a^2=y, a^3 );
                                 $ya$ 

> subs( x=y, y=x, [x,y] );
                                [x, x]

> subs( {x=y, y=x}, [x,y] );
                                [y, x]

> subs( y=0, sin(y) );
                                sin(0)

> eval( sin(y), y=0 );
                                0

```

```

> p := x^7+8*x^6+x^2-9;
                                      $p := x^7 + 8x^6 + x^2 - 9$ 

> op(2,p);
                                      $8x^6$ 

> subsop( 2=y, p );
                                      $x^7 + y + x^2 - 9$ 

> subsop( 2=-op(2,p), p );
                                      $x^7 - 8x^6 + x^2 - 9$ 

> subsop( 1=0, p );
                                      $-9 + 8x^6 + x^2$ 

> subsop( 1=1, x*y*z );
                                      $yz$ 

> subsop( 1=NULL, 2=z, 3=y, [x,y,z] );
                                      $[z, y]$ 

> subsop( 0=g, f[a,b,c] );
                                      $g_{a,b,c}$ 

> p := f(x,g(x,y,z),x);
                                      $p := f(x, g(x, y, z), x)$ 

> subsop( [2,3]=w, p );
                                      $f(x, g(x, y, w), x)$ 

> subsop( [2,0]=h, [2,3]=w, 3=a, p );
                                      $f(x, h(x, y, w), a)$ 

> subsop( p );
                                      $f(x, g(x, y, z), x)$ 

```

Сортировка и селекция выражений широко используется в символьных преобразованиях. Функция сортировки меняет порядок расположения членов в выражении (или порядок расположения выражений). Для выполнения сортировки служит функция **sort**, используемая в одной из следующих форм:

sort(L), sort(L, F), sort(A), sort(A, V),

где L - список выражений, F – необязательная булева процедура, A – алгебраическое выражение, V – необязательные дополнительные переменные.

Примеры.

```

> sort([2,1,3]);
                                      $[1, 2, 3]$ 

> sort([2,1,3], `>`);
                                      $[3, 2, 1]$ 

> sort(1+x+x^2);
                                      $x^2 + x + 1$ 

> sort([c,a,d],lexorder);
                                      $[a, c, d]$ 

> sort([a,ba,aaa,aa],length);
                                      $[a, ba, aa, aaa]$ 

```

```
> p := y^3+y^2*x^2+x^3:
sort(p,[x,y]);
```

$$x^2 y^2 + x^3 + y^3$$

```
> sort(p,[x,y], ascending );
```

$$y^3 + x^3 + x^2 y^2$$

```
> sort(p,[x,y], plex);
```

$$x^3 + x^2 y^2 + y^3$$

```
> sort(p,[x,y], plex, descending );
```

$$x^3 + x^2 y^2 + y^3$$

```
> sort((y+x)/(y-x),x);
```

$$\frac{x+y}{-x+y}$$

Функция **select** служит для выделения нужного выражения.

Примеры.

```
> integers := [$10..20];
```

$$integers := [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]$$

```
> select(isprime, integers);
```

$$[11, 13, 17, 19]$$

```
> remove(isprime, integers);
```

$$[10, 12, 14, 15, 16, 18, 20]$$

```
> selectremove(isprime, integers);
```

$$[11, 13, 17, 19], [10, 12, 14, 15, 16, 18, 20]$$

```
> f := 2*exp(a*x)*sin(x)*ln(y);
```

$$f := 2 e^{(ax)} \sin(x) \ln(y)$$

```
> select(has, f, x);
```

$$e^{(ax)} \sin(x)$$

```
> remove(has, f, x);
```

$$2 \ln(y)$$

```
> selectremove(has, f, x);
```

$$e^{(ax)} \sin(x), 2 \ln(y)$$

```
> f := indets(f);
```

$$f := \{a, x, y, e^{(ax)}, \sin(x), \ln(y)\}$$

```
> select(type, f, 'name');
```

$$\{a, x, y\}$$

```
> remove(type, f, 'name');
```

$$\{e^{(ax)}, \sin(x), \ln(y)\}$$

```
> selectremove(type, f, 'name');
```

$$\{a, x, y\}, \{e^{(ax)}, \sin(x), \ln(y)\}$$

```
> f := 2*ln(x)*(y+1);
```

$$f := 2 \ln(x) (y + 1)$$

```
> c := remove(has, f, x);
```

$$c := 2y + 2$$

```
> f/c;
```

$$2 \frac{\ln(x)(y+1)}{2y+2}$$

```
> select(has, f, x);
```

$$\ln(x)$$

Упрощение выражений.

Функция **simplify** является одной из самых главных в системе символьных вычислений. Она предназначена для упрощения математических выражений и используется в следующих формах:

simplify(expr) – возвращает упрощенное выражение **expr** или повторяет его, если в рамках правил системы Maple упрощение невозможно;

simplify(expr, n1, n2, ...) – возвращает упрощенное выражение **expr** с учетом параметров с именами **n1, n2, ...** (в том числе заданных списком или множеством);

simplify(expr, assume=prop) – возвращает упрощенное выражение **expr** с учетом всех условий.

Возможности функции **simplify** большие. Она обеспечивает упрощение математических выражений, выполняя следующие типовые действия:

- комбинирует цифровые подвыражения;
- приводит подобные множители в произведениях;
- приводит подобные множители в суммах;
- использует тождества, содержащие ноль;
- использует тождества, содержащие единицу;
- распределяет целочисленные показатели в произведениях;
- сокращает **expr** на наибольший общий множитель;
- понижает степень полиномов там, где это возможно;
- использует преобразования, способные упростить выражение.

Тем не менее, функция **simplify** не всегда способна выполнить возможные упрощения (все же это «железо»). В этом случае желательно ей подсказать в какой области ищутся упрощения и где можно найти упрощающие преобразования. В качестве параметров могут задаваться тождества, имена специальных математических функций и указания на область действия упрощений.

Примеры.

```
> simplify(4^(1/2)+3);
```

$$5$$

```
> simplify(exp(a+ln(b*exp(c))));
```

$$b e^{(a+c)}$$

```
> simplify(sin(x)^2+ln(2*x)+cos(x)^2);
```

$$1 + \ln(2) + \ln(x)$$

```
> simplify(sin(x)^2+ln(2*x)+cos(x)^2, trig);
```

$$1 + \ln(2x)$$

```
> simplify(sin(x)^2+ln(2*x)+cos(x)^2, ln);
```

$$\sin(x)^2 + \ln(2) + \ln(x) + \cos(x)^2$$

```
> e := cos(x)^5 + sin(x)^4 + 2*cos(x)^2 - 2*sin(x)^2 - cos(2*x):  
simplify(e);
```

$$\cos(x)^4 (\cos(x) + 1)$$


```

> f := -1/3*x^5*y + x^4*y^2 + 1/3*x*y^3 + 1:
simplify(f, {x^3 = x*y, y^2 = x+1});

$$1 + y^5 + y^4 - 2y^3 - y^2 + y$$


> g:=sqrt(x^2);

$$g := \sqrt{x^2}$$


> simplify(g);

$$\operatorname{csgn}(x) x$$


> simplify(g,assume=real);

$$|x|$$


> simplify(g,assume=positive);

$$x$$


> simplify(g,symbolic);

$$x$$


> simplify(sin@arcsin, `@`);

$$(\ ) \rightarrow \operatorname{args}$$


> simplify(exp@ln, `@`);

$$(\ ) \rightarrow \operatorname{args}$$


> invfunc[f] := g:
simplify(f@g@@2);

$$f@(g^{(2)})$$


> simplify(hypergeom([-1],[1],z), hypergeom);

$$1 - z$$


> simplify(hypergeom([ ],[3], ln(z)), hypergeom);

$$2 \frac{\operatorname{BesselI}(2, 2\sqrt{\ln(z)})}{\ln(z)}$$


> simplify(WhittakerW(7/3,1, hypergeom([1],[2],z)),hypergeom);

$$\operatorname{WhittakerW}\left(\frac{7}{3}, 1, -\frac{e^z(-1 + e^{(-z)})}{z}\right)$$


> simplify(GAMMA(n+1)/GAMMA(n),GAMMA);

$$n$$


> simplify(GAMMA(n+1)*(n^2+3*n+2),GAMMA);

$$\Gamma(n+3)$$


> e := [24^(1/3), 24^(-1/3), (-24)^(1/3)];

$$e := \left[ 24^{(1/3)}, \frac{1}{24} 24^{(2/3)}, (-24)^{(1/3)} \right]$$


> simplify(e,radical);

$$\left[ 2 \cdot 3^{(1/3)}, \frac{1}{6} 3^{(2/3)}, (1 + \sqrt{3} I) 3^{(1/3)} \right]$$


> e := [6^(1/3)+25^(1/3), 6^(1/3)+15^(1/3)];

$$e := [6^{(1/3)} + 25^{(1/3)}, 6^{(1/3)} + 15^{(1/3)}]$$


> simplify(e,radical);

```

$$[6^{(1/3)} + 5^{(2/3)}, 3^{(1/3)} 2^{(1/3)} + 3^{(1/3)} 5^{(1/3)}]$$

> **e := (-8*b^3*a)^(1/3);**

$$e := (-8 b^3 a)^{(1/3)}$$

> **simplify(e,radical);**

$$2 (-b^3 a)^{(1/3)}$$

> **simplify(e,radical,symbolic);**

$$2 b (-a)^{(1/3)}$$

> **assume(b<0);**

simplify(e,radical);

$$-2 b \sim a^{(1/3)}$$

> **f := (x^4+3*x^3*y+3*x^2*y^2+x*y^3)^(1/3);**

$$f := (x^4 + 3 x^3 y + 3 x^2 y^2 + x y^3)^{(1/3)}$$

> **simplify(f,radical);**

$$(x (x + y)^3)^{(1/3)}$$

> **simplify(f,radical,symbolic);**

$$(x + y) x^{(1/3)}$$

> **f := (2*x+2)^(1/3) + (4*x+4)^(1/3) + (3*x-3)^(1/3);**

$$f := (2 x + 2)^{(1/3)} + (4 x + 4)^{(1/3)} + (3 x - 3)^{(1/3)}$$

> **simplify(f,radical);**

$$(2 x + 2)^{(1/3)} + 2^{(1/3)} (2 x + 2)^{(1/3)} + (3 x - 3)^{(1/3)}$$

> **f := (x+1)^(4/3)-x*(x+1)^(1/3);**

$$f := (x + 1)^{(4/3)} - x (x + 1)^{(1/3)}$$

> **simplify(f,radical);**

$$(x + 1)^{(1/3)}$$

Расширение выражений.

Иногда требуется расширить выражение (например, раскрыть скобки в произведении или разложить рациональную дробь на простые дроби, ...). Для этого можно использовать функцию **expand**. Она имеет следующую форму:

expand(expr, expr1, ..., exprn),

где **expr** – расширяемое выражение, **expr1, ..., exprn** – необязательные выражения – опции.

Примеры.

> **expand((x+1)*(x+2));**

$$x^2 + 3 x + 2$$

> **expand((x+1)/(x+2));**

$$\frac{x}{x + 2} + \frac{1}{x + 2}$$

> **expand(1/(x+1)/x);**

$$\frac{1}{(x + 1) x}$$

```

> expand(sin(x+y));
sin(x) cos(y) + cos(x) sin(y)

> expand(cos(2*x));
2 cos(x)^2 - 1

> expand(exp(a+ln(b)));
e^a b

> expand((x+1)*(y+z));
x y + x z + y + z

> expand((x+1)*(y+z), x+1);
(x+1) y + (x+1) z

> expand(ln(x/(1-x)^2));
ln( x / (1-x)^2 )

> assume(x,real);
expand(ln(x/(1-x)^2));
ln( 1 / (1-x)^2 ) + ln(x)

> expand(BesselJ(2,t));
2 BesselJ(1,t) / t - BesselJ(0,t)

> expand(LegendreQ(2,t));
-1/4 ln(t+1) + 1/4 ln(t-1) + 3/4 t^2 ln(t+1) - 3/4 t^2 ln(t-1) - 3/2 t

> expand(JacobiP(3,1,1/4,t));
-307/1024 - 2247/1024 t + 1575/1024 t^2 + 5075/1024 t^3

```

Разложение на множители (факторизация).

Для разложения алгебраического выражения на множители используется функция факоризации:

factor(a), factor(a, K),

где **a** – полином от нескольких переменных и **K** – необязательный параметр.

Примеры.

```

> factor(6*x^2+18*x-24);
6 (x+4) (x-1)

> factor(6);
6

> ifactor(6);
(2) (3)

> factor((x^3-y^3)/(x^4-y^4));
y^2 + y x + x^2 / ((x+y) (y^2 + x^2))

> factor(x^3+5);
x^3 + 5

```

```

> factor(x^3+5, 5^(1/3));

$$(x^2 - x 5^{(1/3)} + 5^{(2/3)})(x + 5^{(1/3)})$$

> factor(x^3+5, {5^(1/3), (-3)^(1/2)});

$$\frac{1}{4}(2x - 5^{(1/3)} - \sqrt{-3} 5^{(1/3)})(2x - 5^{(1/3)} + \sqrt{-3} 5^{(1/3)})(x + 5^{(1/3)})$$

> factor(x^3+5.0);

$$(x + 1.709975947)(x^2 - 1.709975947x + 2.924017740)$$

> factor(x^3+5, complex);

$$(x + 1.709975947)(x - 0.8549879733 + 1.480882610I)(x - 0.8549879733 - 1.480882610I)$$

> factor(y^4-2, sqrt(2));

$$(y^2 + \sqrt{2})(y^2 - \sqrt{2})$$

> alias(alpha = RootOf(x^2-2));
factor(y^4-2, alpha);

$$(y^2 + \alpha)(y^2 - \alpha)$$

> factor(x^3+y^3);

$$(x + y)(x^2 - yx + y^2)$$

> factor(x^3+y^3, (-3)^(1/2));

$$\frac{1}{4}(2y - x + x\sqrt{-3})(2y - x - x\sqrt{-3})(x + y)$$


```

Комплектование по степеням.

Для комплектования выражения **a** по степеням указанного фрагмента **x** используется функция **collect** в одной из следующих форм:

collect(a, x), collect(a, x, form, func).

Здесь параметр **form** может иметь два значения: recursive и distributed. Параметр **func** позволяет задать имя функции, по которой будет идти комплектование.

Примеры.

```

> f := a*ln(x)-ln(x)*x-x;

$$f := a \ln(x) - \ln(x)x - x$$

> collect(f, ln(x));

$$(a - x) \ln(x) - x$$

> g := int(x^2*(exp(x)+exp(-x)), x);

$$g := x^2 e^x - 2x e^x + 2e^x - \frac{x^2}{e^x} - 2\frac{x}{e^x} - \frac{2}{e^x}$$

> collect(g, exp(x));

$$(2 + x^2 - 2x) e^x + \frac{-2x - 2 - x^2}{e^x}$$

> f := x*(x+1)+y*(x+1);

$$f := x(x + 1) + y(x + 1)$$

> coeff(f, x, 1);

$$1 + y$$

> g := collect(f, x);

```

$$g := x^2 + (1+y)x + y$$

```
> coeff(g,x,1);
```

1+y

Построение графиков.

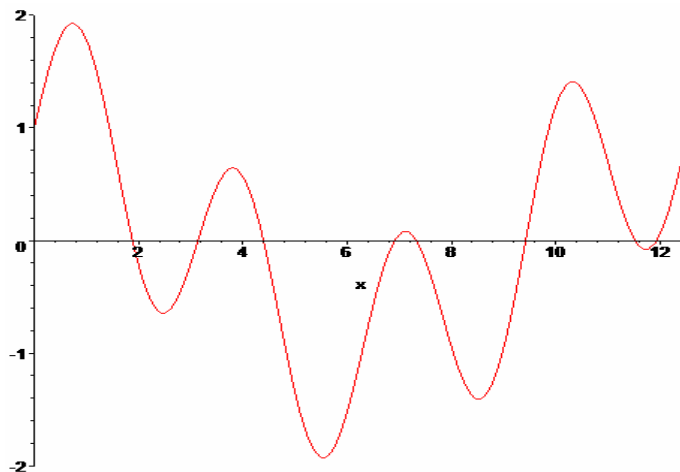
Для построения двумерного графика используется функция **plot**. Она задается в виде:

Plot(f, h, v), Plot(f, h, v, o),

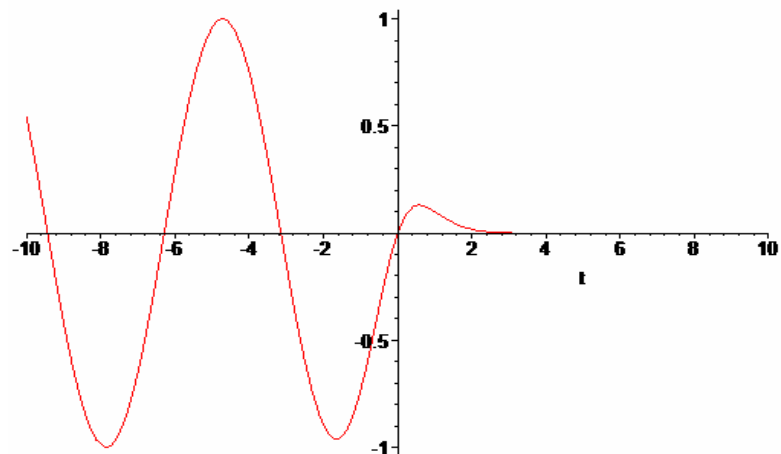
где **f** - функция, **h** - переменная с указанием области изменения, **v** – необязательная переменная с указанием области изменения, **o** - параметр или набор параметров, задающих стиль построения графика (толщина и цвет кривой, тип кривых и т.д.).

Примеры.

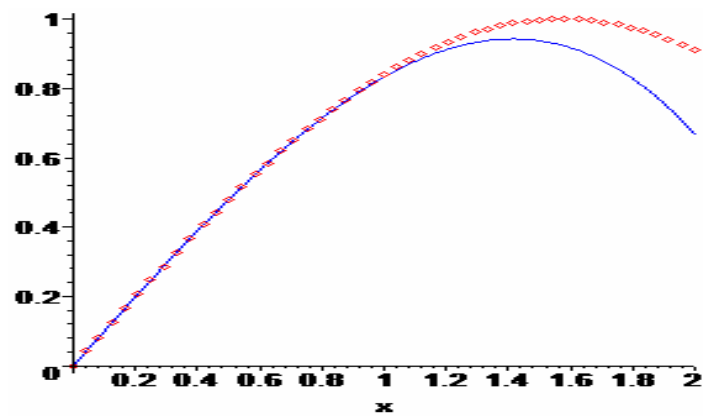
```
> plot(cos(x/2) + sin(2*x), x = 0..4*Pi);
```



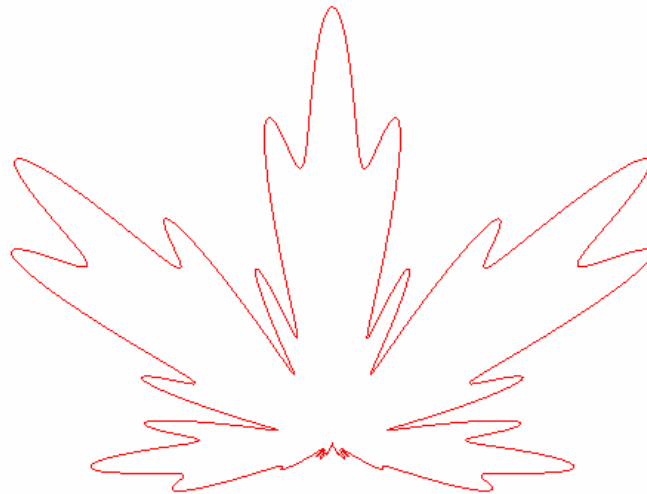
```
> plot(sin(t)/(1 + exp(t)^2), t);
```



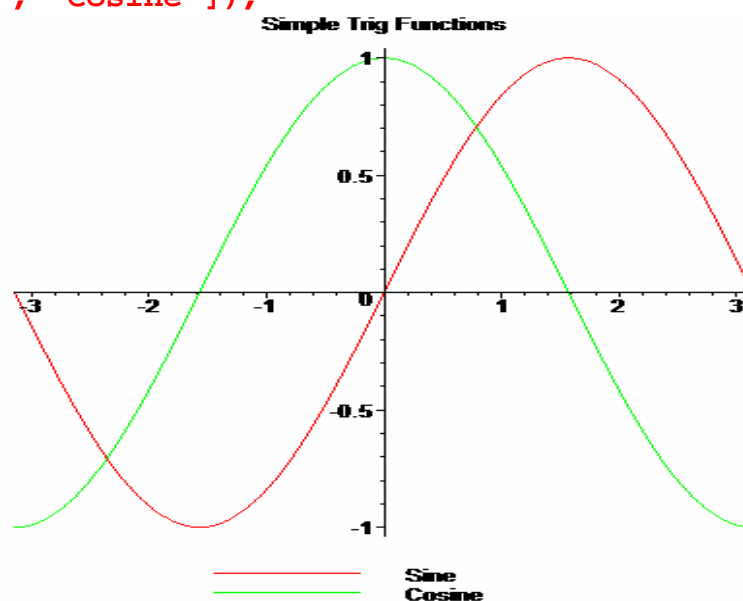
```
> plot([sin(x), x-x^3/6], x=0..2, color=[red,blue],
style=[point,line]);
```



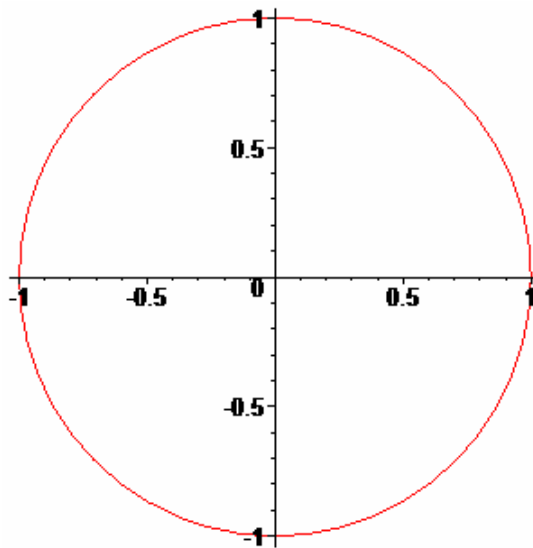
```
> s := t->100/(100+(t-Pi/2)^8): r := t -> s(t)*(2-sin(7*t)-
cos(30*t)/2):
plot([r(t),t,t=-
Pi/2..3/2*Pi],numpoints=2000,coords=polar,axes=None);
```



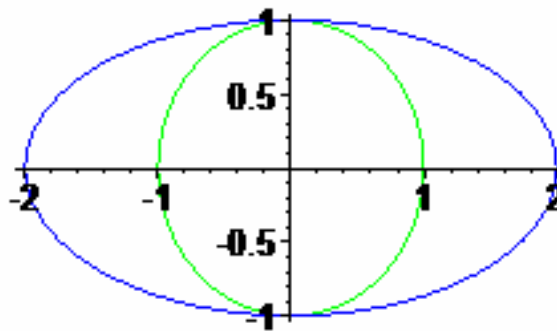
```
> plot([sin, cos], -Pi..Pi, title="Simple Trig Functions",
legend=["Sine", "Cosine"]);
```



```
>
> plot([1,t,t=0..2*Pi],coords=polar);
```



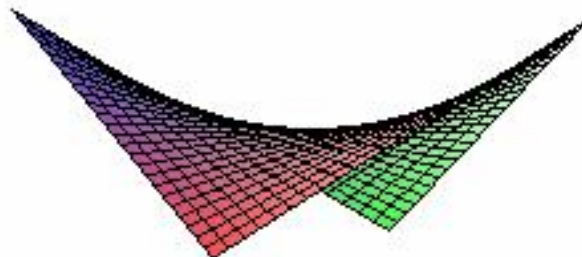
```
> plot([2*cos(s), sin(s), s=0..2*Pi], [cos(t), sin(t),  
t=0..2*Pi]),  
color=[blue,green]);
```



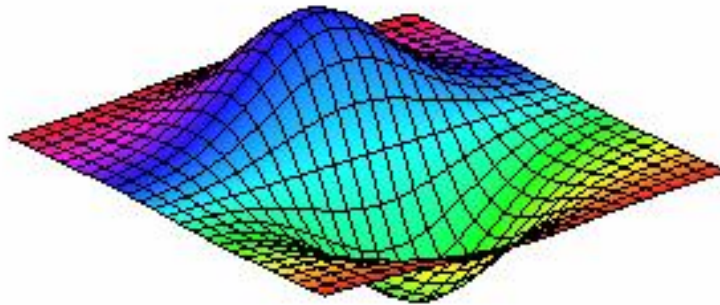
Для построения графиков 2-мерных поверхностей (3-мерный график) используется функция **plot 3d**.

Примеры.

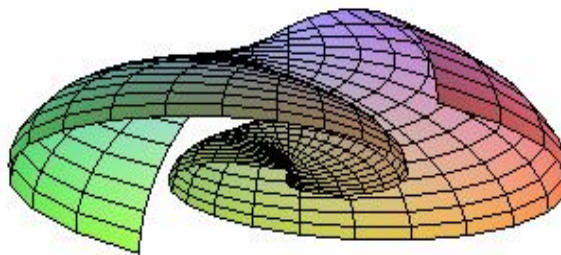
```
> plot3d(x*y, x=-Pi..Pi, y=-Pi..Pi);
```



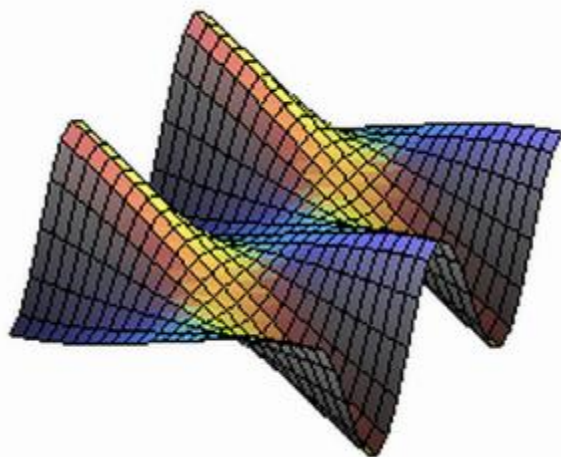
```
> plot3d(x*exp(-x^2-y^2), x=-2..2, y=-2..2, color=x);
```



```
> plot3d([x*sin(x)*cos(y), x*cos(x)*cos(y), x*sin(y)], x=0..2*Pi,
y=0..Pi);
```



```
> plot3d(cos(x)*sin(y), x=-2*Pi..2*Pi, y=-1..1, shading=zgrayscale,
lightmodel=light2);
```



В заключение этого краткого обзора системы Maple отметим, что некоторые пакеты и некоторые возможности системы Maple остались нерассмотренными. Для дальнейшего изучения этой системы рекомендуется литература, указанная в конце. Рекомендуется также изучать справочную систему Maple и, разумеется, применять систему Maple на практике.

Далее рассмотрим другой математический пакет - **Scientific Work Place**.

Часть 2. Система Scientific Work Place

Введение

Название этого пакета переводится, как Научное Рабочее Место. Как это следует из названия, в этом пакете присутствует все, что требуется специалисту в естественных науках, использующего в своей работе математические вычисления, читающего и пишущего научную литературу.

Действительно, пакет **SWP** представляет собой отличный текстовый редактор, специализированный для набора математических, физических и других научных работ любой степени сложности. Кроме этого в пакет встроена система символьных и численных вычислений, с возможностью визуализации этих вычислений. Эта система основана на ядре системы Maple. Интерфейс этой системы вычислений отличается от интерфейса системы Maple и приближен к естественной – большинство формул записывается обычным образом (так же, как в научной литературе и мелом на доске). Однако, эти формулы в SWP – «живые». Достаточно выделить уравнение (элементарное или дифференциальное или матричное), выражение для функции, ... и нажать соответствующую кнопку (или на пункт в контекстном меню) и будет предьявлено точное (символьное) или приближенное решение или построен график. Наконец, в SWP встроена издательская система TeX и результат работы в этом текстовом редакторе сохраняется, как стандартный TeX-файл (точнее, LaTeX-файл) или PDF/PS- файл. В последних версиях SWP предусмотрена возможность взаимного конвертирования этого LaTeX-файла в форматы HTM, RTF и (что особенно ценно) DOC (формат редактора MsWord).

Оп интерфейса SWP.

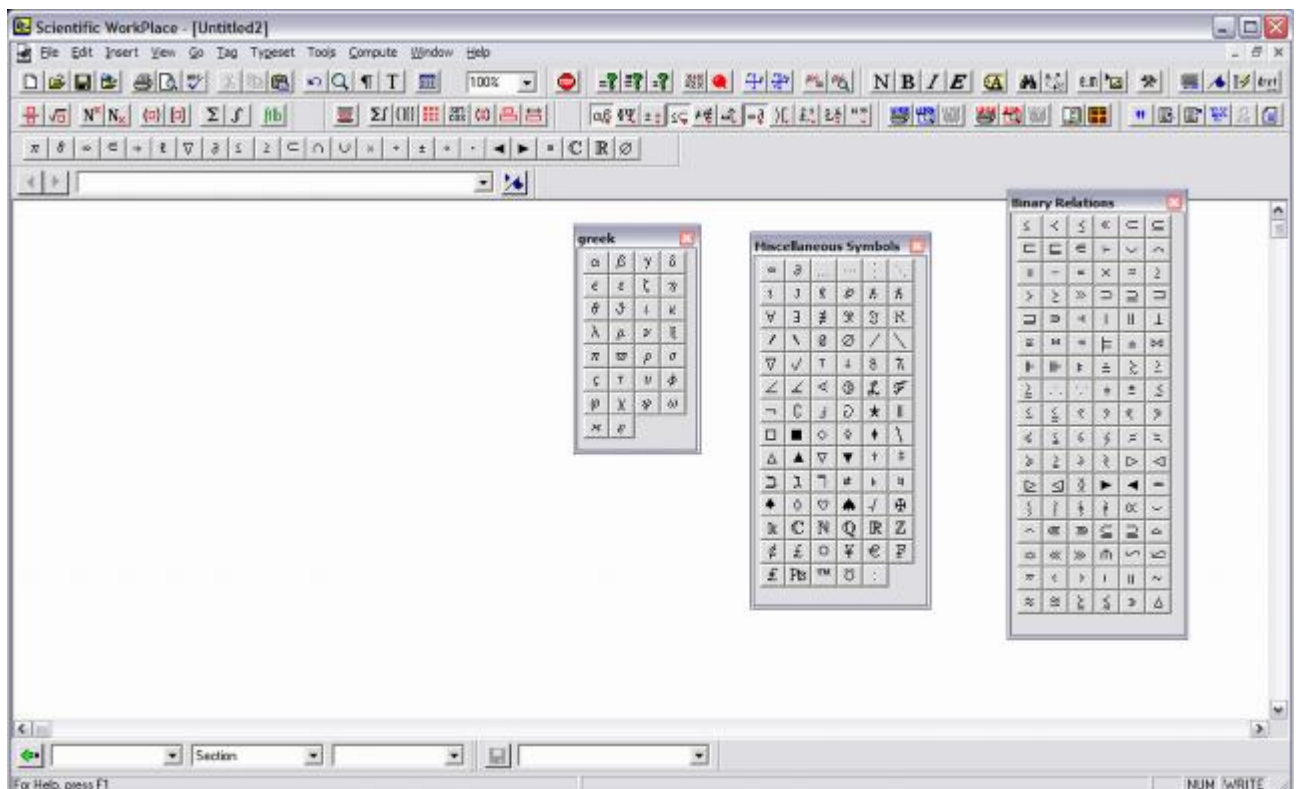


Рис. 1.1 Окно системы SWP

Как у всех приложений под Windows интерфейс SWP имеет ряд характерных элементов, видимых на рис. 1.1 и перечисленных ниже;

- строка заголовка;
- строка главного меню;
- панели инструментов;

контекстные панели инструментов;
окно ввода и редактирования документа;
строка состояния.

Пользовательский интерфейс SWP позволяет готовить документы, содержащие одновременно текстовую информацию, математические формулы, результаты математических действий в естественной математической форме, графические данные и таблицы, содержащие перечисленные элементы.

В основе пользовательского интерфейса SWP лежит графический многооконный интерфейс Windows. Управление SWP происходит с помощью главного меню, панелей инструментов, «горячих» клавиш и мыши.

2. Главное меню.

Наиболее полные возможности управления предоставляет главное меню системы SWP. Оно предоставляет доступ к основным операциям и параметрам интерфейса SWP. Ниже перечисляются меню, доступные при работе с документом:

- * **File** – работа с файлами, просмотр и печать документов;
- * **Edt** – команды редактирования документов и операции с буфером обмена;
- * **Insert** – переключение режимов «математика – текст», вставка математических и иных символов, выключенных формул;
- * **View** – переключение режимов работы с невидимыми символами, матрицами, таблицами, векторами, добавление или удаление панелей инструментов, изменение масштаба просмотра документа;
- * **Go** – переход к другой главе, другому параграфу, отмеченному месту в тексте, к формуле;
- * **Tag** – изменение стиля оформления документа, «горячих клавиш»;
- * **Typeset** – оформление титульной страницы, авторов, аннотации, выбор способа оформления библиографии, способа TeX-компиляции, просмотра и печати документа;
- * **Tools** – некоторые установки пользователя для данного документа;
- * **Compute** – запуск команд системы символьных и численных вычислений;
- * **Window** – работа с окнами;
- * **Help** – работа со справочными материалами.

Многие из перечисленных выше меню понятны большинству пользователей, работающих в Windows. Поэтому рассмотрим меню **Compute**, как наиболее интересное для пользователя, знакомящегося с SWP.

Меню **Compute** включает в себя простые пункты и сложные пункты (т.е. пункты, имеющие свои дополнительные меню). К простым пунктам относятся следующие:

Evaluate (вычислить),
Evaluate Numerically (вычислить приближенно),
Simplify (упростить),
Factor (разложить на множители),
Expand (раскрыть скобки),
Check Equality (проверить равенство),
Power Series (разложение функции в степенной ряд).

К сложным пунктам относятся следующие:

Combine (преобразование в тождественную форму). Дополнительное меню имеет вид:

- + **Exponential**;
- + **Logs**;
- + **Powers**;
- + **Trig Functions**;
- + **Arctan**;
- + **Hiperbolic Trig Function**.

Rewrite (преобразование в тождественную форму другого типа). Дополнительное меню:

- + **Rational** (преобразование типа действительного числа в тип рационального числа);
- + **Float** ((преобразование типа рационального числа в тип действительного числа);

- + **Exponential** (преобразование функции с помощью показательных функций);
- + **Factorial** (преобразование целого числа с помощью факториалов);
- + **Gamma** (преобразование с помощью Гамма-функции);
- + **Logarithm** (аналогично);
- + **Sin and Cos**;
- + **Sinh and Cos**;
- + **Tan**;
- + **Sin**;
- + **Cos**;
- + **Arcsin**;
- + **Arccos**;
- + **Arctan**;
- + **Arccot**;
- + **Polar** (преобразование комплексного числа в показательную форму);
- + **Rectangular** (аналогично);
- + **Normal Form**;
- + **Equation as Matrix** (переход от системы уравнений к расширенной матрице);
- + **Matrix as Equation** (обратный переход).

Solve (решить или вычислить). Дополнительное меню имеет вид:

- + **Exact** (точно или символично);
- + **Numeric** (численно или приближенно);
- + **Integer** (в целых числах);
- + **Recursion** (рекурсивно).

Polynomials (работа с полиномами). Дополнительное меню:

- + **Collect** (приведение подобных членов);
- + **Divide** (деление многочленов);
- + **Partial Fractions**;
- + **Roots**;
- + **Sort**;
- + **Companion Matrix**.

Calculus (дифференциальное и интегральное исчисление). Дополнительное меню:

- + **Integrate by Parts** (интегрирование по частям);
- + **Change Variables** (замена переменных);
- + **Partial Fractions** (разложение на простые дроби);
- + **Approximate Integral** (частичная сумма Римана);
- + **Plot Approximate Integral** (график ступенчатой функции в сумме Римана)
- + **Plot Approximate Integral Animated**;
- + **Find Extrema** (найти экстремумы);
- + **Iterate** (итерации функции);
- + **Implicit Differentiation** (вычисление производной неявной функции).

Solve ODE (решение обыкновенного дифф. уравнения). Дополнительное меню:

- + **Exact** (точное (символьное) решение);
- + **Numeric** (численное решение);
- + **Laplace** (с помощью преобразования Лапласа);
- + **Series** (с помощью степенных рядов).

Transforms (интегральные преобразования). Дополнительное меню:

- + **Fourier** (преобразование Фурье);
- + **Inverse Fourier** (обратное преобразование Фурье);
- + **Laplace** (преобразование Лапласа);
- + **Inverse Laplace** (обратное преобразование Лапласа).

Vector Calculus (операции теории поля). Дополнительное меню:

- + **Gradient** (вычисление градиента);

- + **Divergence** (вычисление дивергенции);
- + **Curl** (вычисление ротора);
- + **Laplacian** (вычисление лапласиана);
- + **Jacobian** (вычисление якобиана);
- + **Hessian** (вычисление гессиана);
- + **Wronskian** (вычисление вронскиана);
- + **Scalar Potential** (вычисление скалярного потенциала);
- + **Vector Potential** (вычисление векторного потенциала);
- + **Set Basis Variables** (определение независимых переменных).

Matrices (работа с матрицами). Дополнительное меню:

- + **Adjugate** (присоединенная матрица);
- + **Characteristic Polynominal** (характеристический многочлен);
- + **Cholesky Decomposition**;
- + **Column Basis** (базис из столбцов);
- + **Concatenate** (объединение матриц);
- + **Condition Number** (число обусловленности);
- + **Definiteness Tests** (тест на определенность);
- + **Determinant** (определитель);
- + **Eigenvalues** (собственные числа);
- + **Eigenvectors** (собственные векторы);
- + **Fill Matrix**;
- + **Fraction free Gaussian Elimination**;
- + **Gaussian Elimination**;
- + **Hermitian Normal Form** (эрмитова нормальная форма);
- + **Hermitian Transpose** (эрмитова сопряженная матрица);
- + **Inverse Hermitian** (обратная матрица);
- + **Jordan Form** (жорданова нормальная форма);
- + **Minimum Polynomial** (минимальный многочлен);
- + **Norm** (норма матрицы);
- + **Null Space Basis** (базис ядра);
- + **Orthogonality Test** (тест на ортогональность);
- + **Permanent** (неалгебраическая сумма произведений элементов);
- + **PLU Decomposition**;
- + **QR Decomposition**;
- + **Random Matrix** (случайная матрица);
- + **Rank** (ранг матрицы);
- + **Rational Canonical Form**;
- + **Reduce Row Echelon Form**;
- + **Reshape**;
- + **Row Basis** (базис из строк);
- + **Singular Values** (сингулярные числа);
- + **Smith Normal Form** (обычная форма агента Смита);
- + **Spectral Radius** (спектральный радиус);
- + **Stack**;
- + **SVD** (сингулярные векторы);
- + **Trace** (сумма диагональных элементов);
- + **Transpose** (транспонированная матрица).

Simplex (симплекс метод). Дополнительное меню:

- + **Dual**;
- + **Feasible**;
- + **Maximize**;
- + **Minimize**;

+ **Standardize/**

Statistics (математическая статистика). Дополнительное меню:

- + **Fit Curve to Data** (кривая данных);
- + **Random Numbers** (генерация случайных чисел (12 способов));
- + **Mean** (математическое ожидание);
- + **Median**;
- + **Mode**
- + **Correlation**;
- + **Covariance**;
- + **Geometric Mean**;
- + **Harmonic Mean**;
- + **Mean Deviation**;
- + **Moment**;
- + **Quantile**;
- + **Standard Deviation**;
- + **Variance**.

Plot 2D (двумерный график). Дополнительное меню:

- + **Rectangular** (график в декартовой системе координат);
- + **Polar** (график в полярной системе координат);
- + **Implicit** (график неявной функции);
- + **Parametric** (график функции, заданной параметрически);
- + **Conformal** (образ координатной сетки при конформном отображении);
- + **Vector Field** (построение поля направлений);
- + **ODE** (график численного решения обыкновенного дифф. уравнения).

Plot 3D (трехмерный график). Дополнительное меню:

- + **Rectangular** (график в декартовой системе координат);
- + **Cylindrical** (график в цилиндрической системе координат);
- + **Spherical** (график в сферической системе координат);
- + **Implicit** (график неявной функции);
- + **Tube** (график трубчатой поверхности – раздутия заданной кривой);
- + **Gradient** (векторное поле – градиент функции);
- + **Vector Field** (построение векторного поля).

Plot 2D Animated (построение семейства кривых). Дополнительное меню:

- + **Rectangular** (семейство кривых в декартовой системе координат);
- + **Polar** (семейство кривых в полярной системе координат);
- + **Implicit** (семейство кривых, заданных неявно);
- + **Parametric** (семейство кривых, заданных параметрически);
- + **Conformal** (образ координатной сетки при семействе отображений);
- + **Vector Field** (визуализация семейства векторных полей);
- + **Gradient** (векторное поле – градиент семейства функций);
- + **ODE** (семейство интегральных кривых).

Plot 3D Animated (построение семейства поверхностей). Дополнительное меню:

- + **Rectangular** (семейство поверхностей в декартовой системе координат);
- + **Cylindrical** (семейство поверхностей в цилиндрической системе координат);
- + **Spherical** (семейство кривых в сферической системе координат);
- + **Implicit** (семейство поверхностей, заданных неявно);
- + **Tube** (анимация раздутия кривых);
- + **Gradient** (векторное поле – градиент семейства поверхностей);
- + **Vector Field** (семейство векторных полей).

Definitions (определения символов, операторов и функций). Дополнительное меню:

- + **New Definition**;
- + **Undefined**;

- + **Show Definitions;**
- + **Clear Definitions;**
- + **Restore Definitions;**
- + **Define MuPAD Name.**

Settings;

Interpret/


3. Панели инструментов.


В интерфейсе SWP насчитывается 17 панелей. Перечислим их.


3.1. Standard toolbar (стандартная панель).




Сюда входят следующие кнопки.

 **File/ New** (создать документ)


 **File/ Open** (открыть документ)

 **File / Save** (сохранить документ)

 **File / Open Location** (открыть Интернет документ)

 **File / Print** (печать документа без спецформатирования)

 **File / Preview** (просмотр документа без спецформатирования)

 **Tools / Spelling** (проверка правописания)

 **Edit / Cut** (вырезать)

 **Edit / Copy** (копировать)

 **Edit / Paste** (вставить)

 **Edit / Undo** (отменить последнее действие)

 **Edit / Properties** (свойства объекта)

 **View / Invisibles** (показать / скрыть невидимые символы)

  **Insert / Math or Text** (режимы математика/ текст)

 **Insert / Table** (вставить таблицу)

 **View / Custom** (масштаб)


3.2. Stop toolbar (кнопка прекращения вычислений)





3.3. Typeset toolbar (LaTeX, PDF-установки работы с документом)



Сюда входят следующие кнопки.

 **Typeset / Print** (печать TeX-документа)

 **Typeset / Preview** (просмотр TeX-документа)

 **Typeset / Compile** (создание TeX-документа)

 **Typeset / Print PDF** (печать PDF-документа)

 **Typeset / Preview PDF** (просмотр PDF-документа)

 **Typeset / Compile PDF** (создание PDF-документа)

 **Typeset / Front Matter** (редактирование титула)

 **Typeset / Options and Packages** (опции документа)

3.4. Typeset Object Toolbar (LaTeX-редакция документа)



Сюда входят следующие кнопки.



Insert / Typeset Object / Citation (ссылка)



Insert / Typeset Object / Index Entry (оглавление)



Insert / Typeset Object / Cross Reference (перекрестная ссылка)



Insert / Typeset Object / TeX Field (поле TeX команды)



Insert / Typeset Object / Subdocument (поддокумент)



Insert / Typeset Object / Bibliography (библиография)

3.5. Math Templates toolbar (математическая панель №1)



Insert / Fraction (дробь)



Insert / Radical (дробь)



Insert / Superscript (верхний индекс)



Insert / Subscript (нижний индекс)



Insert / Brackets (круглые скобки)




Insert / Fraction (квадратные скобки)

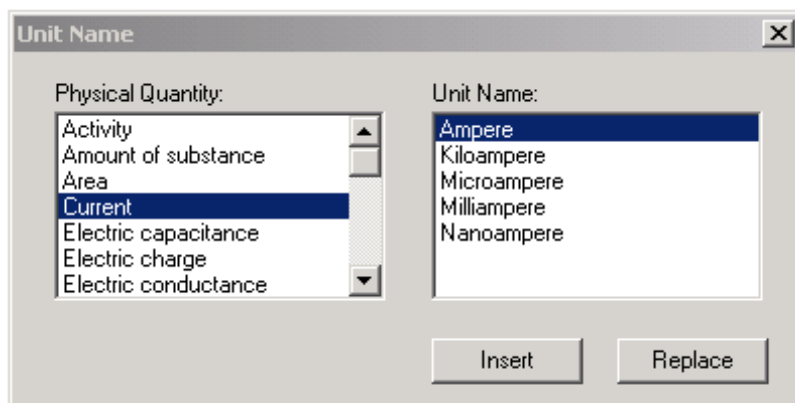


Insert / Operator (сумма)

 **Insert / Operator** (интеграл)

 **Insert / Unit Name** (вычисляемая единица измерения)


После нажатия этой кнопки появляется меню



3.6. Math Objects toolbar (математическая панель №2)

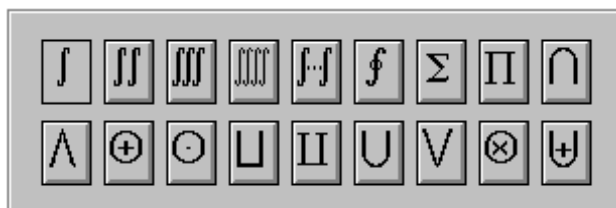


Сюда входят следующие кнопки.

 **Insert / Display** (выключенная формула)

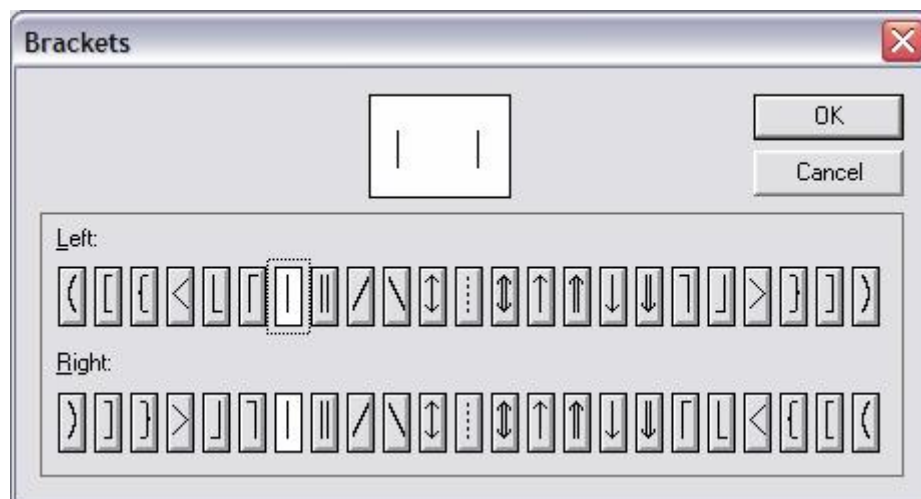
 **Insert / Operator** (сумма, произведение, интеграл)

После нажатия этой кнопки появляется меню



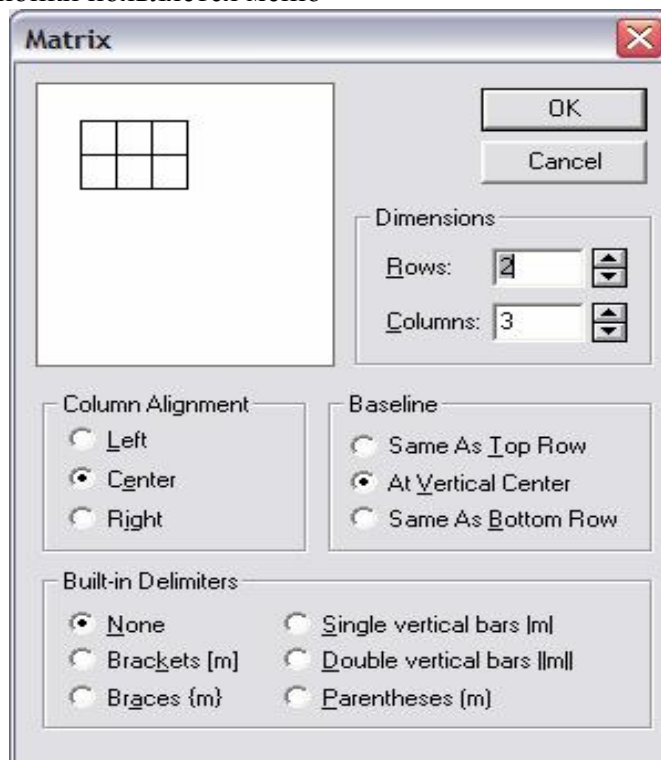
 **Insert / Brackets** (варианты скобок)

После нажатия этой кнопки появляется меню



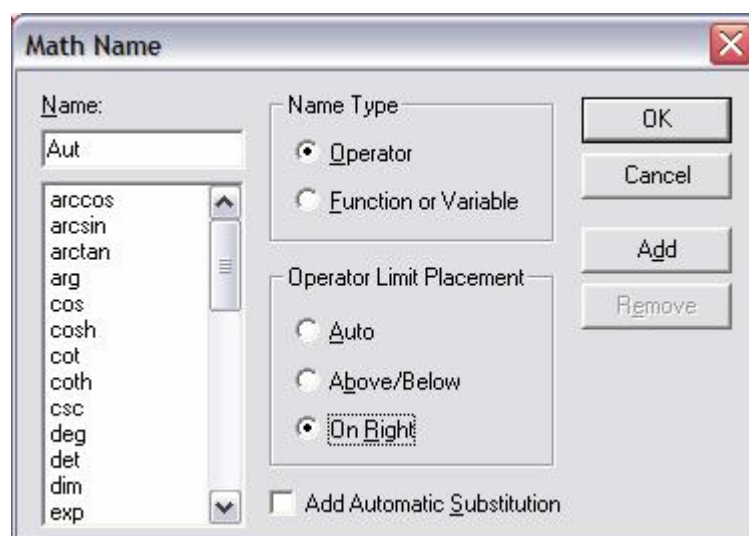
Insert / Matrix (матрица или вектор)

После нажатия этой кнопки появляется меню



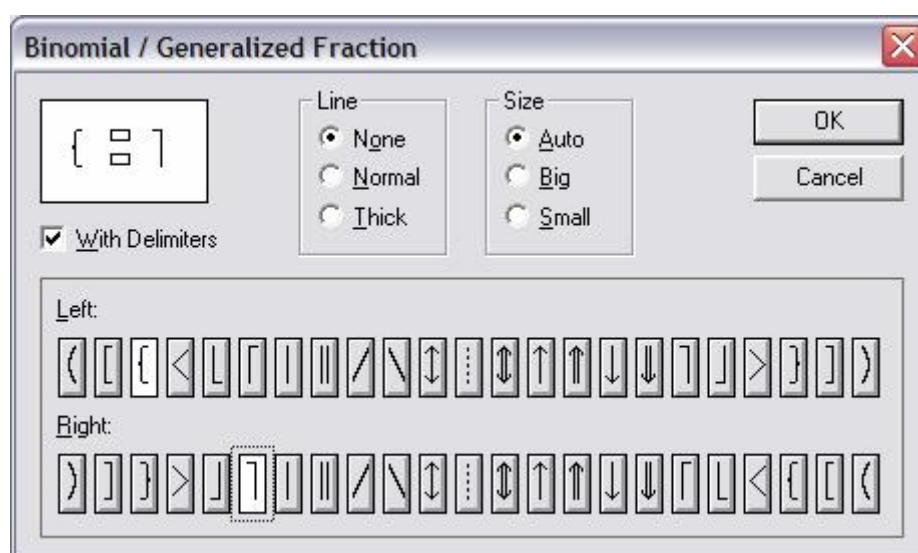
Insert / Math Name (функция)


После нажатия этой кнопки появляется меню



 **Insert / Binomial** (биномиал, обобщенная дробь)

После нажатия этой кнопки появляется меню



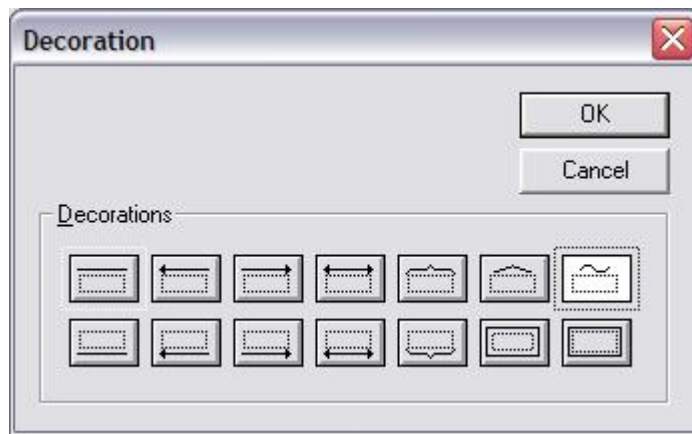
 **Insert / Label** (индекс для выражения)

После нажатия этой кнопки появляется меню



 **Insert / Decoration** (украшение)

После нажатия этой кнопки появляется меню



3.7. Symbol Panels Toolbar (панель символов)



Сюда входят следующие кнопки.



Lowercase Greek (малые греческие буквы)

После нажатия этой кнопки появляется меню



Uppercase Greek (большие греческие буквы)

После нажатия этой кнопки появляется меню



Греческие буквы можно вводить с клавиатуры. После нажатия клавиш Ctrl-g раскладка клавиатуры принимает вид



После введения буквы раскладка клавиатуры восстанавливается. Греческие буквы можно вводить с клавиатуры, используя TeX-команды



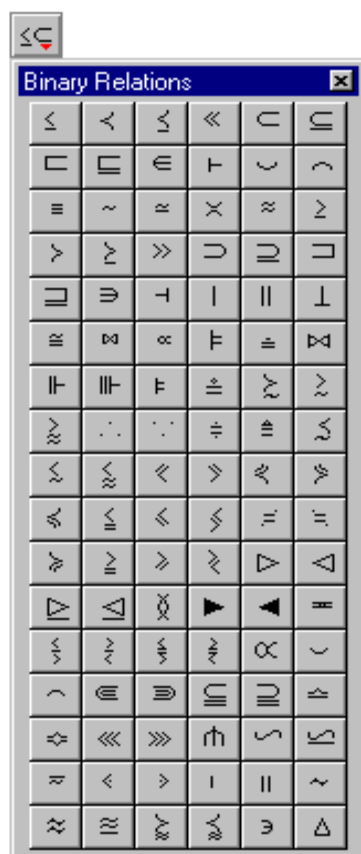
Binary operations (бинарные операции)

После нажатия этой кнопки появляется меню



Binary relations (бинарные отношения)

После нажатия этой кнопки появляется меню



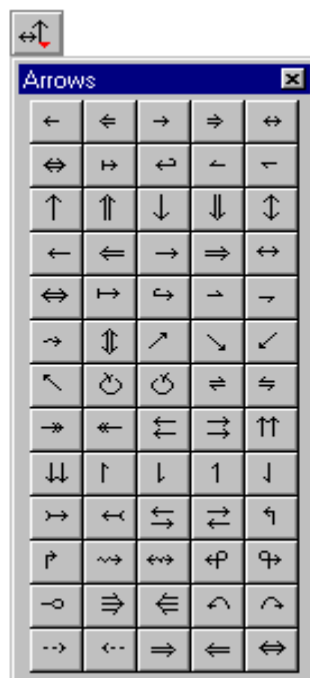
Negated Relations (отрицание отношения)

После нажатия этой кнопки появляется меню



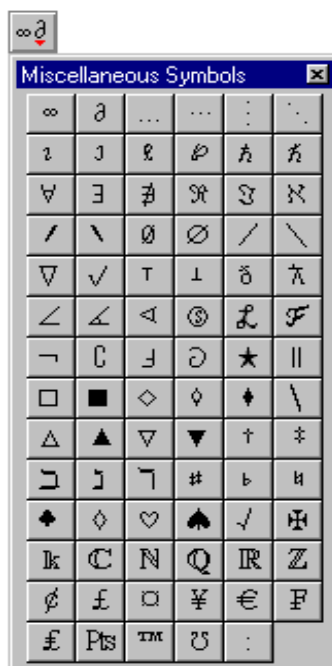
Arrows (стрелки)

После нажатия этой кнопки появляется меню



Miscellaneous Symbols (разные математические символы)

После нажатия этой кнопки появляется меню



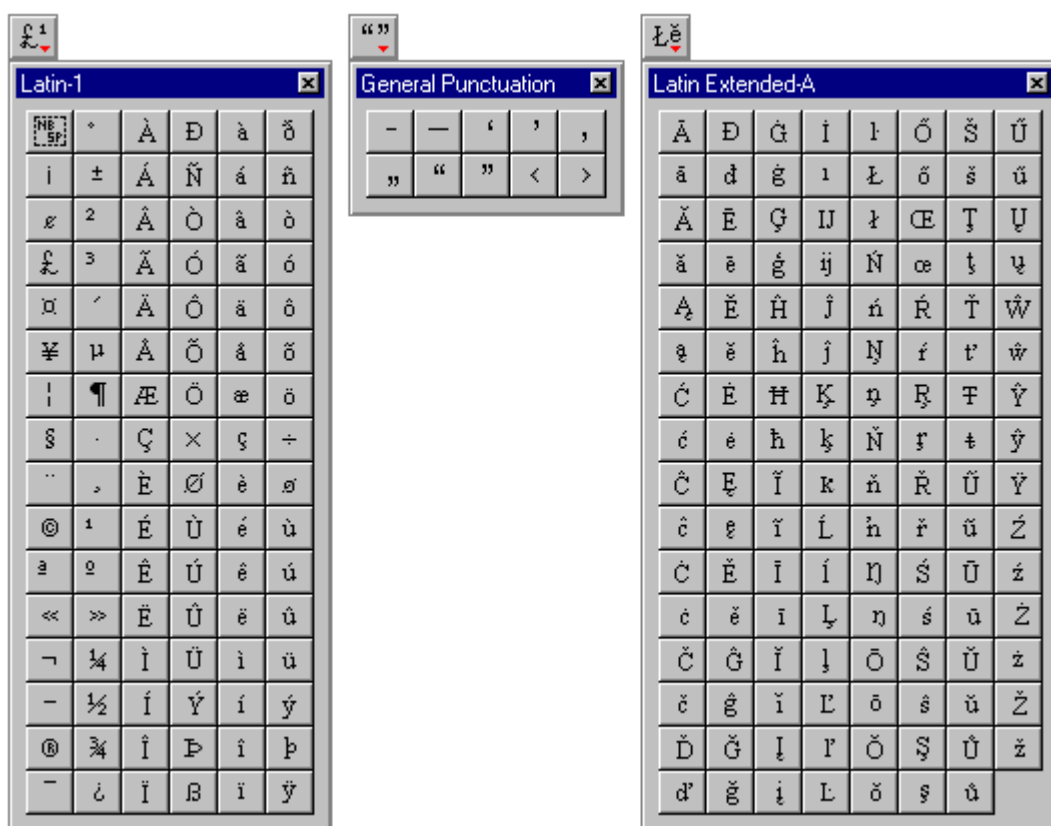
Special Delimiters (ограничители)

После нажатия этой кнопки появляется меню



Unicode Characters (символы **Unicode**)

После нажатия этих кнопок появляется меню



3.8. Compute Toolbar (Панель вычислений).



Сюда входят следующие кнопки.



Compute / Evaluate (вычислить)



Compute / Evaluate Numerically (вычислить приближенно)



Compute / Solve/Exact (решить точно)



Compute / Simplify (упростить)



Compute / Expand (раскрыть скобки)



Compute / Plot 2D / Rectangular (2D график)



Compute / Plot 3D / Rectangular (3D график)



Compute / Define / New Definition (новое определение)



Compute / Define / Show Definitions (показать определения)

3.9. Exam Toolbar (Экзаменационная панель). Она используется для построения и генерирования экзаменационных вопросов, контрольных, домашних и самостоятельных работ и т.д.



Сюда входят следующие кнопки:



View the quiz (просмотр тестовых вопросов)



Tools / Exam Builder (построение тестов)

3.10. Navigate toolbar (Навигационная панель).



Сюда входят следующие кнопки:



Go / Previous Section (перейти к предыдущему разделу)



Go / Next Section (перейти к следующему разделу)



Go / Goto Marker (перейти к отмеченному месту)

3.11. History Toolbar (обзор документа с записью ходов)



Сюда входят следующие кнопки:



Go / History Back (вернуться в начало обзора)

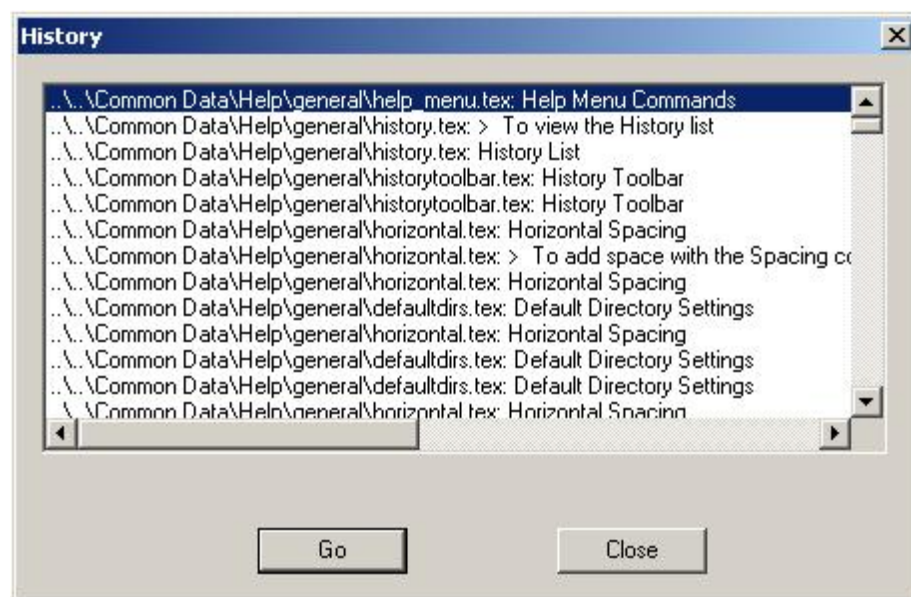


Go / History Forward (отменить **History Back**)



Go / Goto Marker (перейти к отмеченному месту)

При обзоре (**History Back**) создается список просматриваемых мест (**History List**). Для просмотра этого списка и быстрого перехода одному из просмотренных мест в меню **Go** надо выбрать **View History**. Тогда появится следующее окно



3.12. Symbol Cache Toolbar (панель часто используемых символов).



С помощью мышки символ можно перенести на эту панель или удалить.

3.13. Field Toolbar (панель полей).



Сюда входят следующие кнопки:



Insert / Note (вставить сноску, замечание)



Insert / Marker (создать отмеченное место)



Insert / Hypertext Link (вставить гиперссылку)



Insert / Formula (вставить формулу в генерируемый тест)

3.14. Editing Toolbar (панель редактирования).



Сюда входят следующие кнопки:



Tag / Apply / (Normal) (обычный текст)



Tag / Apply / Bold (жирный текст)



Tag / Apply / Italic (курсив)



Tag / Apply / Emphasized (выразительный текст)



Tag / Appearance (изменить элемент стиля (шрифт, ...))



Edit / Find (найти слово, символ, ...)



Edit / Replace (заменить слово, символ, ...)



Insert / Spacing (вставить пробел в формулу)

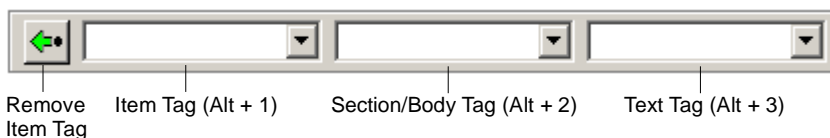


File / Import Picture (вставить графику из файла)



Tools / User Setup (изменить опции пользователя)

3.15. Tag Toolbar (панель тегов (шаблонов))

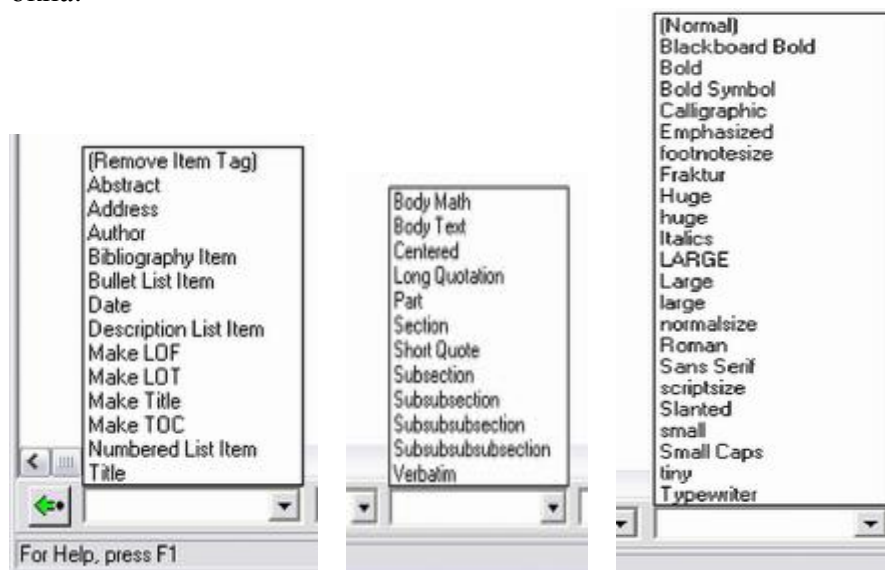


Сюда входят кнопка удаления тега:



Tag / Apply / Remove Item Tag (удалить тег из списка)

и следующие окна:

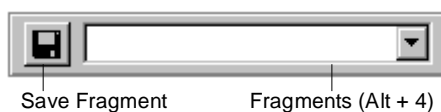


Item Tag (использовать /удалить тег),

Section / BodyTag (стиль заголовков / текста),

Text Tag (тип шрифта)

3.16. Fragments Toolbar (панель фрагментов).



Используется для вставки и сохранения фрагментов (фрагментом называется информация, сохраненная в отдельном файле для быстрой вставки).

3.17. Link Toolbar (панель переходов по содержанию)



Эта панель содержит в себе уже упомянутую панель **History Toolbar**.

Сюда входят следующие кнопки:



Go / Links / Contents (содержание)



Go / Links / Parent Document (исходный раздел)



Go / Links / Beginning Document (рабочий раздел)



Go / Links / Previous Document (предыдущий раздел)



Go / Links / Next Document (следующий раздел)



Go / Links / End Document (последний раздел)



Go / History Back (вернуться в начало)



Go / History Forward (отмена возврата в начало)

Вычисления и построение графиков

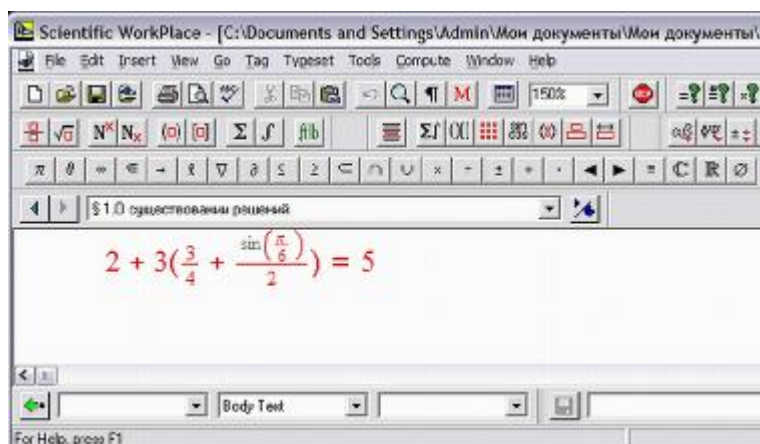
Примеры вычислений в SWP.

Простейшие арифметические вычисления.

Для введения чисел и выполняемых математических символов следует переключатель **T/M** на главной панели привести в положение **M**. Затем набрать формулу и нажать кнопку «вычислить» (evaluate)



на панели вычислений (или, с помощью мыши, выделите выражение и выберите в главном меню Compute/Evaluate; или, выделив выражение, наберите **Ctrl+e**). В следующем рисунке приводится пример.



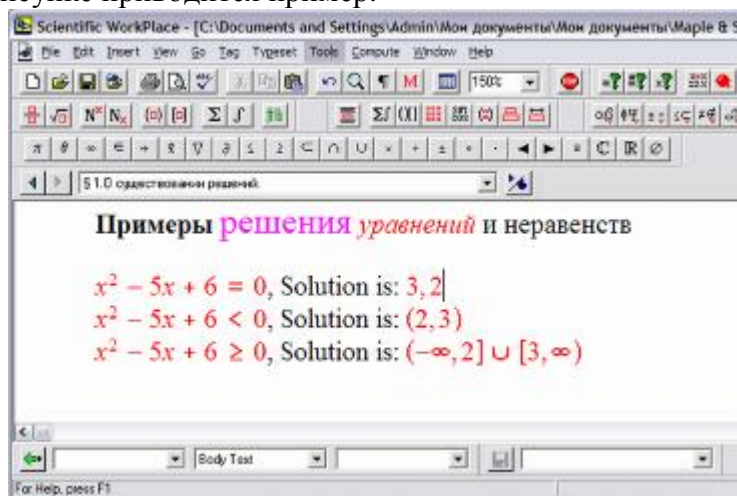
Решение алгебраических уравнений и неравенств.

В режиме **M** наберите уравнение или неравенство и, выделив его, нажмите кнопку



«solve exact» (решить точно) на панели вычислений .

В следующем рисунке приводится пример.



Преобразование выражений.

Упростить. В режиме **M** наберите выражение и, выделив его, нажмите кнопку



«simplify» на панели вычислений.

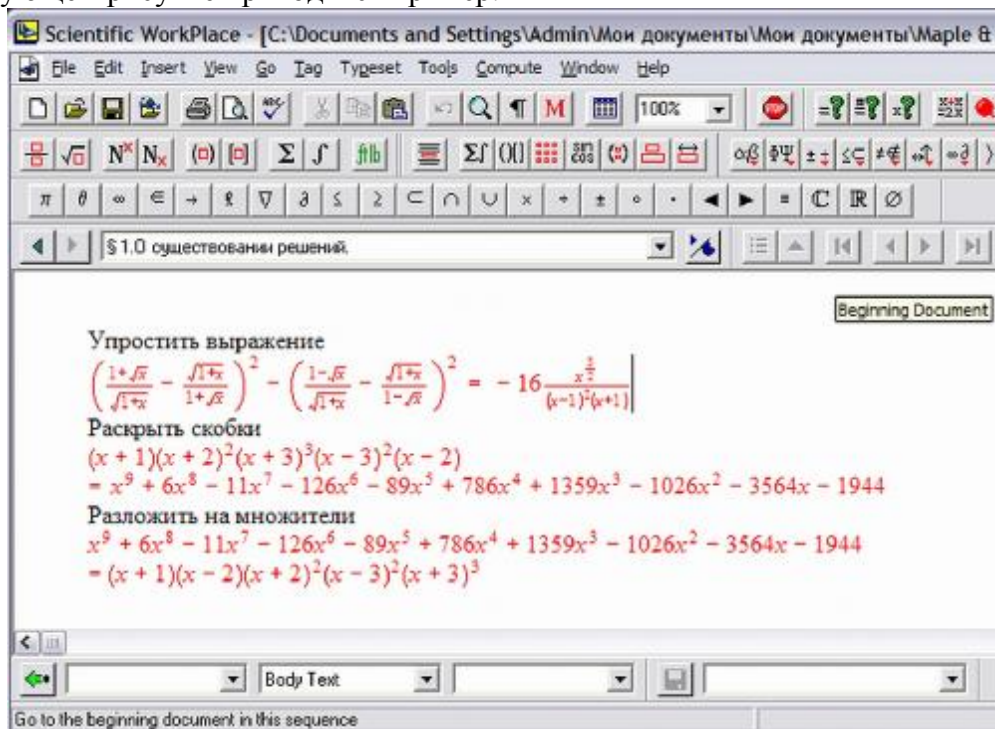
Раскрыть скобки. В режиме **M** наберите выражение и, выделив его, нажмите кнопку







«expand» на панели вычислений.




Разложить на множители. В режиме **M** наберите выражение и, выделив его, в меню **Compute** выберите команду **Factor**.



В следующем рисунке приводится пример.



Вычисления интегралов, производных, пределов и экстремумов.

Для вычисления интеграла надо (в режиме **M**) ввести символ интеграла , пределы интегрирования (если интеграл определенный) с помощью кнопок , , выражение в скобках, дифференциал переменного интегрирования и нажать кнопку .

Для нахождения предела надо (в режиме **M**) ввести символ предела с помощью кнопки **Function** на математической панели , с помощью кнопки  ввести выражение вида $a \rightarrow b$, выражение и нажать кнопку .

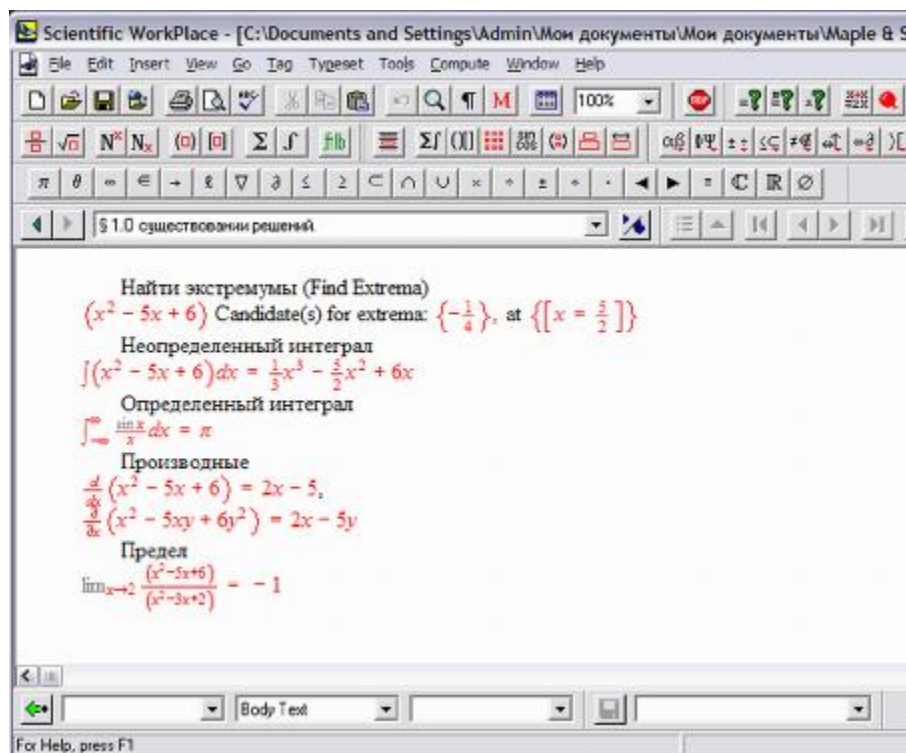
Для нахождения производной надо (в режиме **M**) ввести символ производной $\frac{d}{dx}$ с помощью кнопки  на математической панели, ввести выражение в скобках нажать кнопку .

Для нахождения частной производной следует использовать символ ∂ на панели наличных символов (Symbol Cache Toolbar)



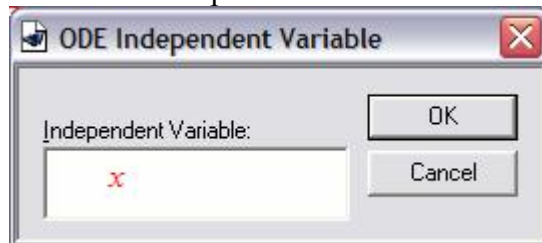
Для нахождения экстремумов функции надо (в режиме **M**) ввести функцию и, выделив его, выбрать в меню **Compute** команду **Find Extrema**.

Пример вычисления интегралов, производных, пределов и экстремумов.



Решение дифференциальных уравнений.

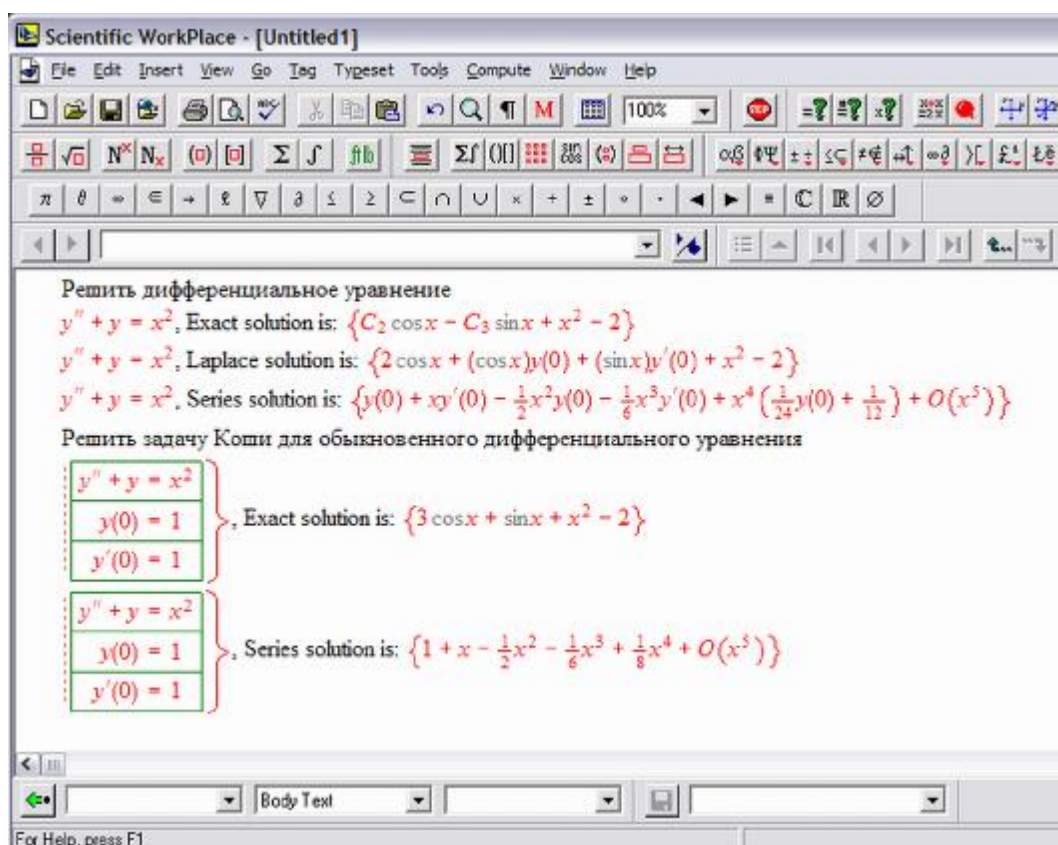
Для решения обыкновенного дифференциального уравнения надо ввести это уравнение, используя для обозначения дифференцирования любой из следующих символов: $\frac{d^n}{dx^n}$ или ' (символ «штрих») и в меню **Compute / Solve ODE** выбрать способ решения. Здесь доступны следующие команды: **Exact** (решить точно), **Laplace** (с помощью преобразования Лапласа), **Numeric** (решить численно) и **Series** (с помощью степенных рядов). После этого возникает окно указания независимого переменного.



Для решения задачи Коши для обыкновенного дифференциального уравнения надо ввести

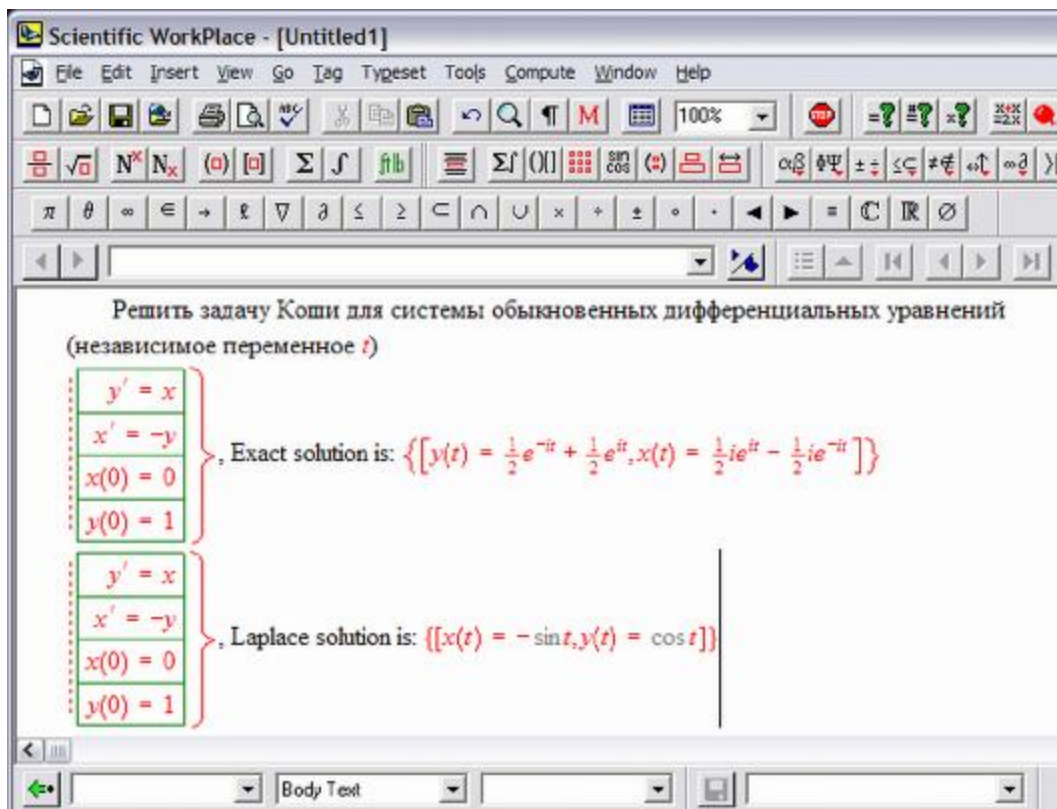


уравнение и начальные условия с помощью кнопки (матрица), где надо выбрать матрицу-вектор из одного столбца, число строк которой равно числу начальных условий +1. В этом столбце надо ввести сначала уравнение (в первую строку), а затем, последовательно, начальные условия. Далее приводятся примеры.



Аналогично решаются и системы обыкновенных дифференциальных уравнений. Здесь также надо ввести уравнения и начальные условия в матрицу-столбец: сначала уравнения, а затем начальные условия.

Пример решения задачи Коши для системы дифференциальных уравнений.




Построение графика решения задачи Коши.

Для построения график численного решения задачи Коши надо ввести задачу Коши, как это было указано выше ($y' = \dots$, $y(0) = \dots$), в меню **Compute/ Solve ODE** выбрать команду **Numeric** (численно). После этого появляется ответ: *Functions defined: y*. Затем, выделив **y** в этом ответе, надо выбрать команду **Compute/Plot 2D/ Rectangular**. Построенный график можно редактировать (см. построение графиков).

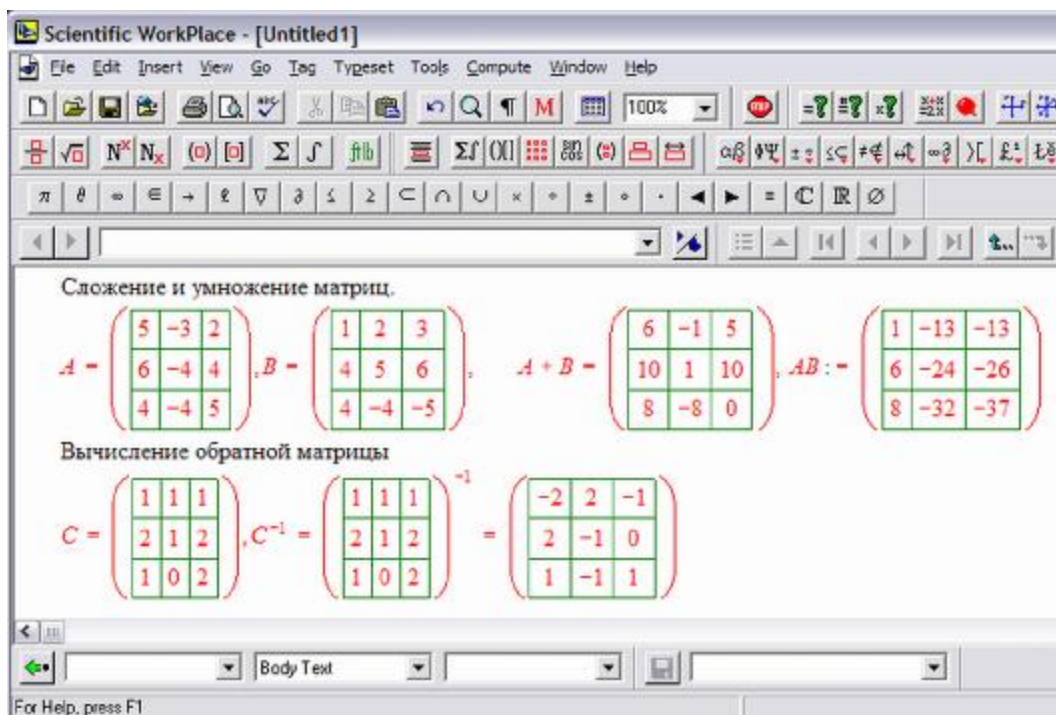
Операции с матрицами.

Действия с матрицами проводятся аналогично действиям с числами и доступны в

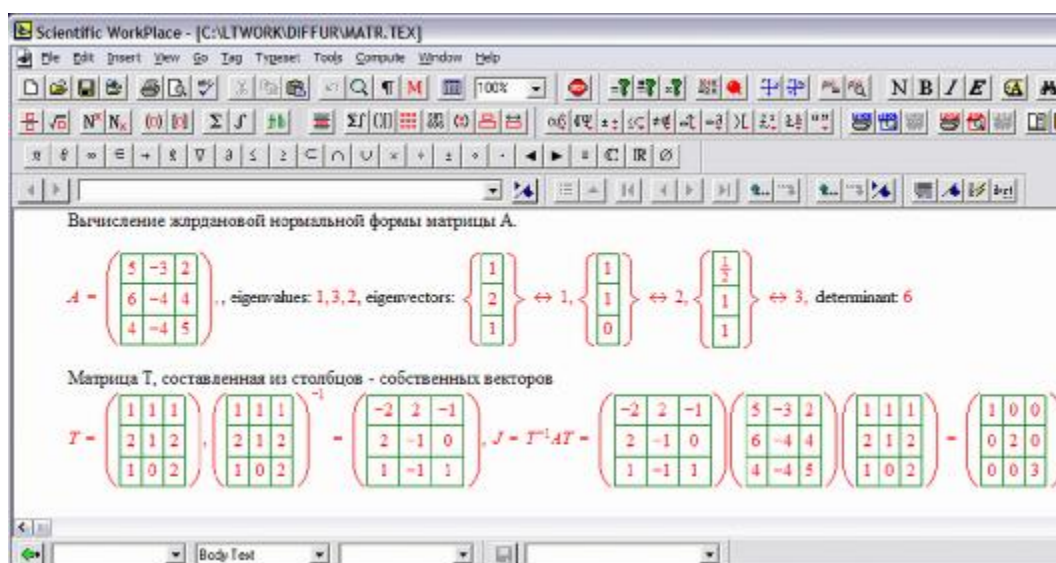
меню **Compute / Matrices**. Матрица вводится с помощью кнопки . Для сложения матриц используется символ +, символ умножения (*) можно опускать (как и при умножении чисел). Для вычисления обратной матрицы можно использовать символ степени (-1),

вводимый с помощью кнопки .

Пример действий с матрицами.



Меню действий с матрицами достаточно большое (доступно 37 команд). Поэтому ограничимся еще одним примером – вычисление жордановой нормальной формы: вычисление собственных чисел и собственных векторов, матрица перехода к жорданову базису, жорданова форма.



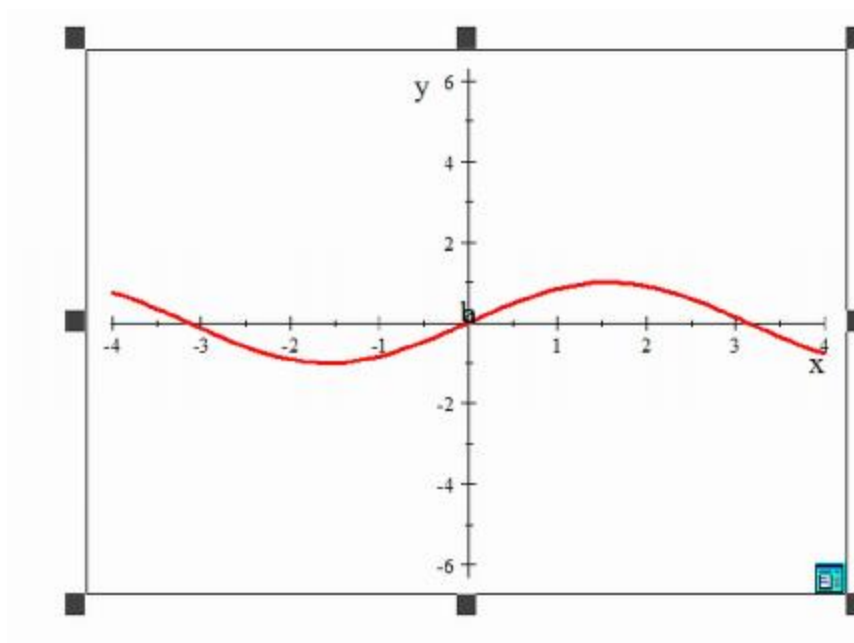
Построение 2D графиков.

В меню **Compute / Plot 2D** среди прочих доступны команды **Rectangular** (график в декартовой системе координат), **Polar** (график в полярной система координат), **Implicit** (график функции, заданной неявно), **Parametric** (график функции, заданной параметрически). Рассмотрим построение графика в декартовой системе координат. Для построения 2D графика функции в декартовой системе координат проще всего ввести выражение этой

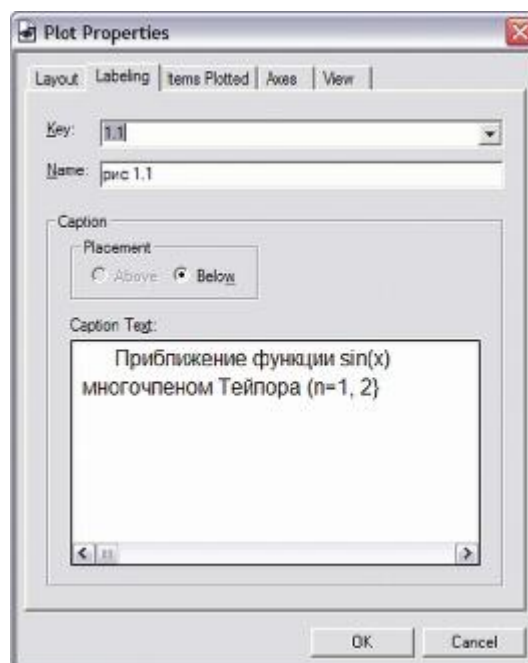
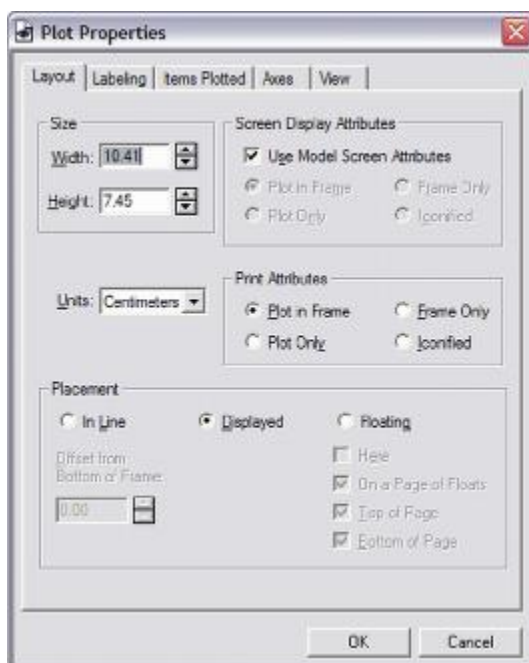


функции (в режиме **M**), затем выделить это выражение и нажать кнопку **2D Plot**. По умолчанию строится график функции в декартовой системе координат (команда **Compute / Plot 2D / Rectangular**).

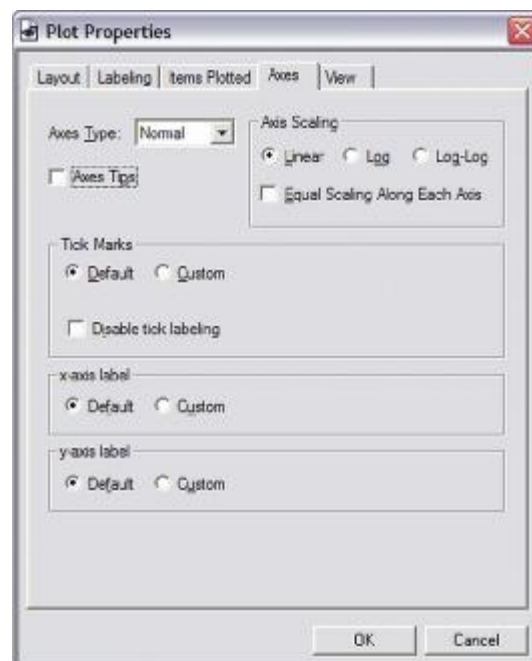
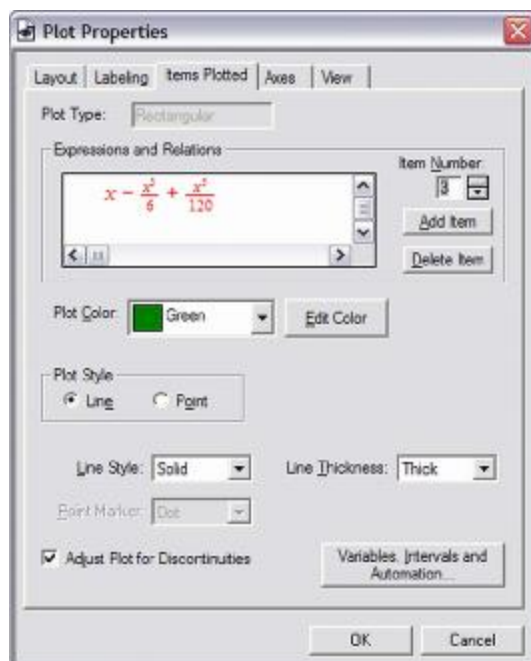
Для примера построим в одном окне графики трех функций: $\sin(x)$, x , $x-x^3/6+x^5/120$. Сначала построим график одной функции - $\sin(x)$.



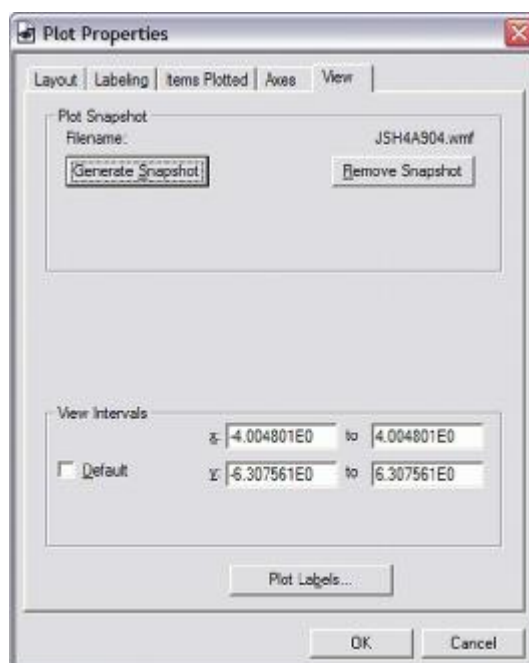
Построенный график можно редактировать. Если нажать на зеленый квадратик в правом нижнем углу графика, то откроется окно **Plot Properties** (свойства графика)



В первом окне устанавливаются размеры графика и его положение. В следующем окне создается подпись к графику и его внутреннее имя в документе для быстрого вызова.

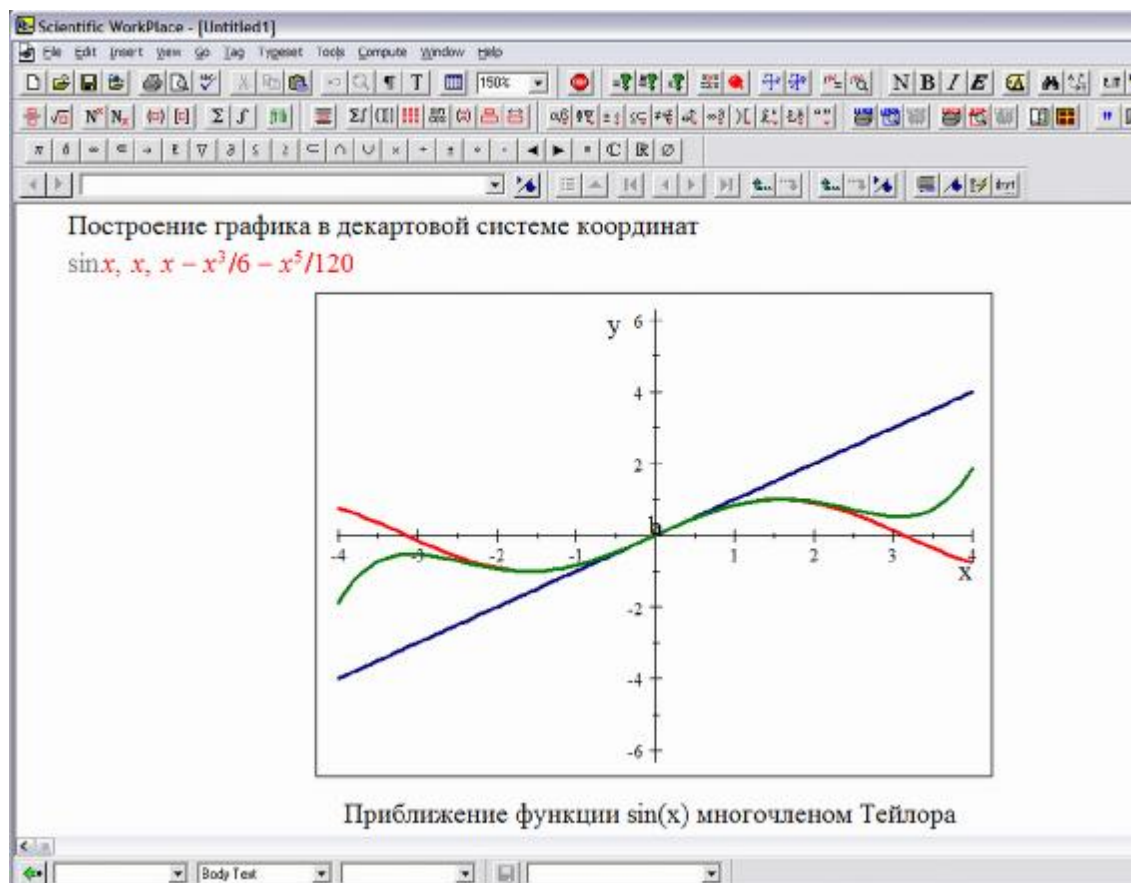


В третьем окне вводится функция и задается тип линии, ее толщина и цвет. И этом же окне можно добавить функции и параметры их графики. В четвертом окне задаются свойства системы координат.

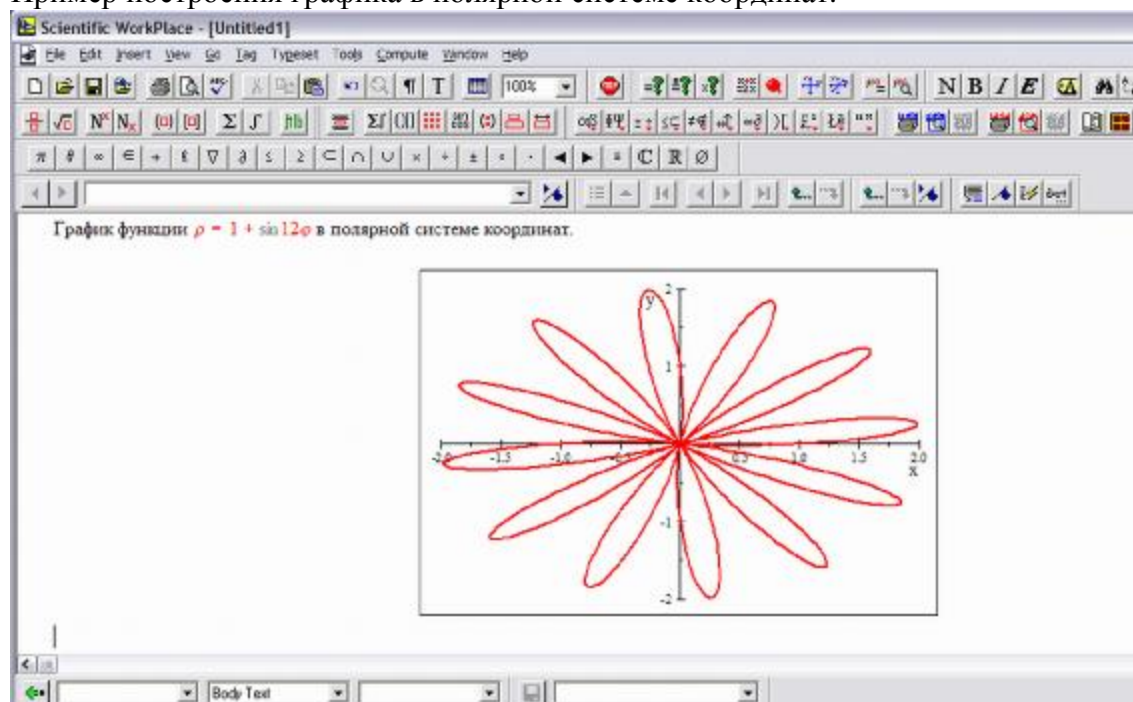


Наконец, в пятом окне задаются область определения и область изменения графиков функций

В результате получился следующий график



Пример построения графика в полярной системе координат.



Сохранение и конвертирование документа. По умолчанию документ сохраняется во внутреннем формате SWP - *.tex. Такой документ не является стандартным TeX-файлом. Для того, чтобы получился стандартный TeX-файл, надо воспользоваться опцией **Save As** и сохранить файл, как **Portable LaTeX** (сокращенный LaTeX).

Этот документ можно конвертировать в форматы *.rtf и *.htm. Для этого надо воспользоваться опцией **Export Document**. Оба этих формата (*.rtf и *.htm) поддерживаются

редактором MsWord. Поэтому TeX-файл можно преобразовать в документ MsWord. и обратно. Для адекватного сохранения формул **весьма желательно**, чтобы в программе MsWord была доступна (т.е. установлена в Windows) программа MathType версии 5 и выше.

Заключение.

Здесь была предпринята попытка краткого введения в математические пакеты Maple и Scientific Work Place. Некоторые возможности системы Maple и некоторые возможности системы SWP остались нерассмотренными. Автор этого обзора надеется, что читатель заинтересуется возможностями систем Maple и SWP и попробует использовать их в своей научной деятельности.

Список литературы

1. Дьяконов В. Maple 7: учебный курс. С-Пб, Питер, 2002, 667с.
2. Говорухин В. Н., Цибулин В.Г. Введение в Maple. М., Мир, 1997, 208 с.
3. Maple. Help system.
4. Scientific Work Place. Help system.