

Министерство образования и науки Российской Федерации
*Федеральное государственное бюджетное образовательное
высшего профессионального образования*

**«АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМ. И.И. Ползунова» (АлтГТУ)**

Естественнонаучный факультет
Кафедра «Физики и технологии композиционных материалов»

Новиковский Е.А.

Учебное пособие

Работа в MathCAD 15



Барнаул 2013

Новиковский, Е. А. Учебное пособие «Работа в системе MathCAD»

[Текст] / Е. А. Новиковский. – Барнаул: Типография АлтГТУ, 2013. – 114 с.

Рассмотрено и одобрено на заседании кафедры физики и
технологии композиционных материалов.

Протокол № 2 от 10.02.13

Данное методическое пособие посвящено описанию программы MathCAD 15.

Пособие структурно состоит из введения, шести лабораторных работ и приложения. Во введении дается краткое описание программы и основные её преимущества. В первой лабораторной работе происходит знакомство обучаемых с интерфейсом программы, основам ввода текста и выражений, простейшими вычислениями. Вторая лабораторная работа посвящена работе с векторами и матрицами. В третьей лабораторной работе рассматриваются методы решений линейных и нелинейных уравнений и их систем, в т.ч. по формулам Крамера и методом Гаусса и символьным методом. В четвертой работе описана методика построения графиков функций и поверхностей, в т.ч. сферической и цилиндрической системе координат. Пятая работа посвящена методике обработки экспериментальных данных. Последняя работа посвящена основам программирования в данной системе.

В конце данного учебного пособия приведены приложения и список рекомендуемой литературы.

Содержание

| | |
|--|----|
| ВВЕДЕНИЕ | 7 |
| Знакомство с программой MathCAD | 9 |
| 1 Интерфейс Mathcad | 9 |
| 1.1 Рабочее окно Mathcad..... | 9 |
| 1.2 Главное меню..... | 10 |
| 1.3 Панели инструментов..... | 11 |
| 2 Ввод, редактирование и форматирование математических выражений | 14 |
| 2.1 Курсоры Mathcad | 14 |
| 2.2 Определение значений переменных | 15 |
| 2.3 Определение функции..... | 16 |
| 2.4 Вычисление выражения | 16 |
| 2.5 Редактирование формул..... | 17 |
| 2.6 Ввод и редактирование текста..... | 19 |
| 2.7 Операции с областями..... | 20 |
| 2.8 Числовые константы. Ввод греческих букв | 22 |
| 2.9 Функции пользователя | 22 |
| 2.10 Дискретные переменные. Построение таблиц | 23 |
| 2.9 Форматирование чисел | 24 |
| 3 Основные арифметические операции | 25 |
| 3.1 Тригонометрические функции | 25 |
| 3.2 Логарифмы | 26 |
| 4 Символьные операции | 27 |
| 4.1 Упрощение выражений | 27 |
| 4.2 Раскрытие выражений..... | 28 |
| 4.3 Разложение на множители | 28 |
| 4.4 Приведение подобных слагаемых | 29 |
| 4.5 Определение коэффициентов полинома..... | 30 |
| 4.6 Вычисление пределов..... | 30 |
| 4.7 Вычисление производной или интеграла | 31 |
| 4.8 Разложение на простейшие дроби..... | 33 |

| | |
|--|----|
| Векторы и матрицы. | 34 |
| 1 Создание массивов | 34 |
| 1.1 Заполнение матрицы по шаблону | 34 |
| 2.2 Изменение размеров матрицы | 35 |
| 2.3 Нумерация элементов матрицы..... | 35 |
| 2.4 Встроенная переменная ORIGIN..... | 36 |
| 2.5 Определение параметров матрицы и векторов | 36 |
| 2.6 Образование новых матриц из уже существующих | 38 |
| 2.7 Ввод больших массивов..... | 39 |
| 2.8 Сортировка векторов и матриц | 40 |
| 2.9 Матричные операторы | 41 |
| 2.9.1 Транспонирование матрицы | 41 |
| 2.9.2 Вычисление определителя матрицы | 41 |
| 2.9.3 Нахождение матрицы, обратной заданной..... | 42 |
| 2.9.4 Сложение, вычитание, умножение и возведение в степень матриц и векторов..... | 42 |
| 2.9.5 Оператор векторизации..... | 45 |
| Решение уравнений | 46 |
| 1 Решение квадратного уравнения..... | 46 |
| 2 Символьное решение уравнений..... | 46 |
| 3 Численное решение уравнений | 47 |
| 4 Поиск экстремума функции..... | 49 |
| 5 Решение систем линейных алгебраических уравнений | 51 |
| 5.1 Решение системы линейных уравнений по формулам Крамера | 52 |
| 5.2 Решение линейной системы методом Гаусса (метод гауссовых исключений)..... | 54 |
| 6 Решения нелинейных уравнений | 55 |
| 6.1 Решение трансцендентных уравнений..... | 55 |
| 6.2 Решение систем нелинейных уравнений при помощи блока «Given – Find» . | 56 |
| 7 Символьное решение систем уравнений | 57 |
| Построение графиков функций и поверхностей | 58 |
| 1 Построение двумерного плоского графика | 58 |
| 1.1 Форматирование двумерных графиков | 60 |

| | |
|--|-----------|
| 1.2 Масштабирование графиков..... | 64 |
| 1.3 Изменение размеров и перемещение графиков | 65 |
| 3 Построение трехмерных графиков | 68 |
| 3.1 Форматирование трехмерных графиков..... | 69 |
| 4 Поверхности тел вращения..... | 70 |
| 5 Построение сложных фигур | 71 |
| 6 Пересекающиеся фигуры | 72 |
| 7 Построение многогранников | 73 |
| 8 График векторного поля | 73 |
| Обработка экспериментальных данных | 75 |
| 1 Функции для обработки экспериментальных данных..... | 75 |
| 1.1 Интерполяция | 76 |
| 1.1.1 Линейная интерполяция..... | 76 |
| 1.1.2 Сплайн-интерполяция | 77 |
| 1.2 Регрессия | 79 |
| 1.2.1 Обобщенная регрессия..... | 81 |
| 1.2.2 Получение аналитического выражения аппроксимирующей функции..... | 85 |
| 1.3 Функции сглаживания..... | 87 |
| Программирование и анимация | 89 |
| 1.1 Создание программ | 89 |
| 1.1.1 Ввод строк в программу..... | 89 |
| 1.1.2. Локальное присвоение значений..... | 90 |
| 1.2 Условный оператор if..... | 91 |
| 1.3 Операторы цикла | 94 |
| 1.3.1 Оператор while..... | 94 |
| 1.3.2 Оператор for | 96 |
| 1.3.3 Операторы break, continue, return | 97 |
| 1.4 Вывод результатов расчета из программы | 98 |
| 1.5 Подпрограммы-функции..... | 98 |
| 1.6 Отладка программ | 99 |
| 1.7 Рекомендации по написанию программ | 100 |
| 2 Анимация | 103 |

| | |
|--|-----|
| 2.1 Создание анимации | 103 |
| ПРИЛОЖЕНИЯ | 106 |
| Приложение 1. Горячие клавиши Mathcad..... | 106 |
| 1 Операторы | 106 |
| 2 Окна и рабочие документы..... | 107 |
| 3 Редактирование..... | 107 |
| Приложение 2. Сообщения об ошибках | 108 |
| Список литературы..... | 114 |

ВВЕДЕНИЕ

Mathcad является уникальной системой для научных и инженерных расчетов и позволяет работать с формулами, числами, текстом и графиками. С помощью системы Mathcad можно решить почти любую математическую задачу символично либо численно. Mathcad позволяет записывать на экране компьютера формулы в их привычном виде.

Mathcad имеет свою собственную справочную систему. Электронные Книги делают доступными для использования в рабочем документе множество полезных формул, справочных данных и диаграмм простым нажатием кнопки.

Объединяя в одном рабочем листе текст, графику, и математические выкладки, Mathcad облегчает понимание самых сложных вычислений.

Особенности Mathcad:

- Полная совместимость с Windows: изменение размеров окон и их перемещение, открытие нескольких окон, поддержка мыши.
- Возможность комбинирования текста, математических выкладок и графики в любом месте экрана.
- Встроенный алгоритм решения систем уравнений и неравенств.
- Действия с производными и интегралами.
- Вычисление сумм рядов, произведений и итерации.
- Тригонометрические, гиперболические, экспоненциальные и Бесселевы функции.
- Статистические функции, включая линейную регрессию, гамма-функцию Эйлера, интеграл ошибок, функции вероятностных распределений.
- Векторы и матрицы, включая операции матричного умножения, обращения, матриц, транспонирования, вычисления определителя матрицы, скалярное и векторное умножение.
- Символьное интегрирование и дифференцирование.
- Обращение, транспонирование и вычисление определителя матриц.
- Многообразие типов графиков: в декартовых координатах, графики в по-

лярных координатах, построение поверхностей в трех измерениях, построение линий уровня, картины векторных полей, трехмерных гистограмм, точечные графики.

- Анимация графиков и любых других объектов рабочего документа.

В данном издании практикуме рассматривается русифицированная версия Mathcad 15. Но для универсальности приведены обозначения пунктов меню, окон и т.п. на английском языке.

В учебном пособии спользуются следующие условные обозначения:

Вставка – обозначение различных меню и окон программы



– предупреждение, следует обязательно обратить внимание



– примечания, рекомендации, советы

[Ctrl] – клавиши и сочетания клавиш на клавиатуре

(:=) – знаки программы

mean – команды и функции программы

Знакомство с программой MathCAD

1 Интерфейс Mathcad

Интерфейс Mathcad по своей структуре аналогичен интерфейсу других Windows- приложений.

1.1 Рабочее окно Mathcad

При открытии файла Mathcad.exe на экране появляется рабочее окно Mathcad с главным меню и пятью панелями: *Standard* (Стандартная), *Formatting* (Форматирование), *Math* (Математическая).

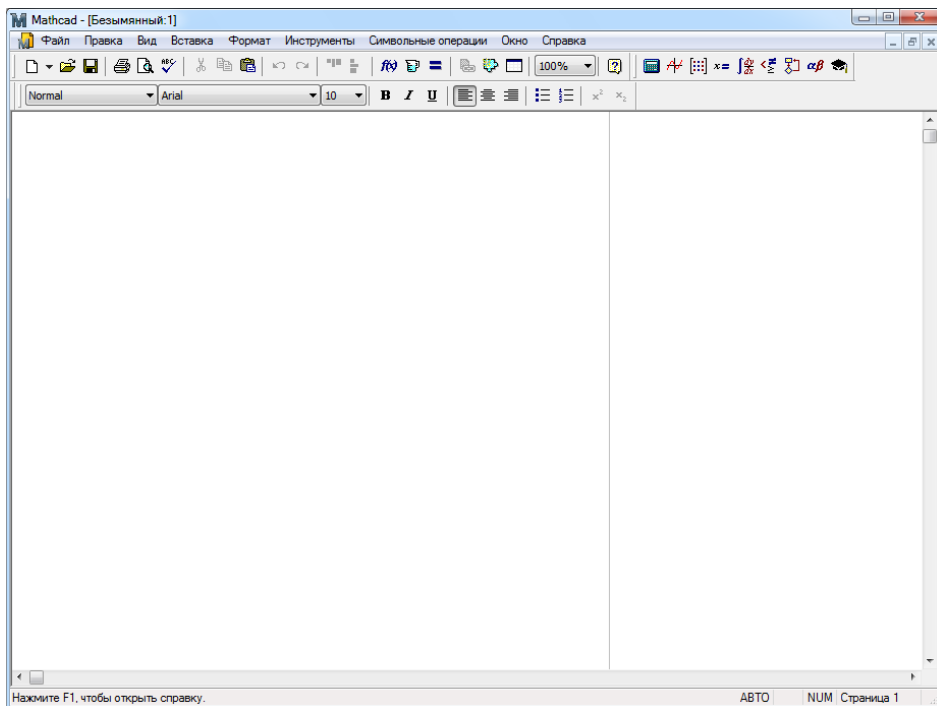


Рисунок 1.1 – Рабочее окно Mathcad

Автоматически загружается файл *Untitled:1* (*Безымянный:1*), представляющий собой шаблон *Normal* (*Обычный*) рабочего документа Mathcad, называемого *Worksheet* (*Рабочий лист*). Кроме того, автоматически загружаются окна *Tip*

of the day (Совет дня) и *Mathcad Resource* (Документация Mathcad). Перед началом работы их надо закрыть. В окне *Tip of the day* следует снять флажок *Show tip at startup* (Показывать совет при загрузке) и щелкнуть на кнопке *Close* (Закрыть).

1.2 Главное меню

Главное меню Mathcad занимает верхнюю строку рабочего окна. Все необходимые действия можно выполнить, следуя пунктам этого меню и последовательно открывающихся окон.

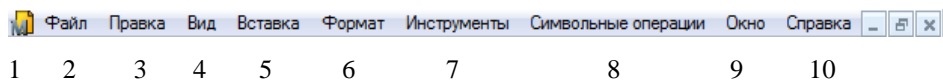


Рисунок 1.2 – Рабочее окно Mathcad

1. Управление рабочим окном Mathcad.
2. *File* (Файл) – команды, связанные с созданием, открытием, сохранением, пересылкой по электронной почте и печатью на принтере файлов с документами.
3. *Edit* (Правка) – команды, относящиеся к правке текста (копирование, вставка, удаление фрагментов и т. д.).
4. *View* (Вид) – команды, управляющие внешним видом документа в рабочем окне Mathcad, а также команды создания файлов анимации.
5. *Insert* (Вставка) – команды вставки различных объектов в документ.
6. *Format* (Формат) – команды форматирования текста, формул и графиков.
7. *Tools* (Инструменты) – команды управления вычислительным процессом.
8. *Symbolics* (Символьные вычисления) – команды символьных вычислений.
9. *Window* (Окно) – команды, позволяющие управлять расположением окон с различными документами на экране.
10. *Help* (Помощь) – команды вызова справочной информации.

1.3 Панели инструментов

Панели инструментов служат для быстрого выполнения наиболее часто применяемых команд. При наведении указателя мыши (курсора) на любую из кнопок рядом с ней появляется всплывающая подсказка – короткий текст, поясняющий назначение кнопки. Содержание панелей (кроме главного меню) можно корректировать. При наведении курсора на панель и нажатии правой кнопки мыши появляется контекстное меню с двумя пунктами, *Hide (Скрыть панель)* и *Customize (Выбрать нужные кнопки)*. Выбрав *Customize*, можно удалить старые и добавить на панель новые кнопки из предлагаемого списка.

- *Standard (Стандартная)* служит для выполнения действий с файлами, редактирования документов, вставки объектов и т. д.

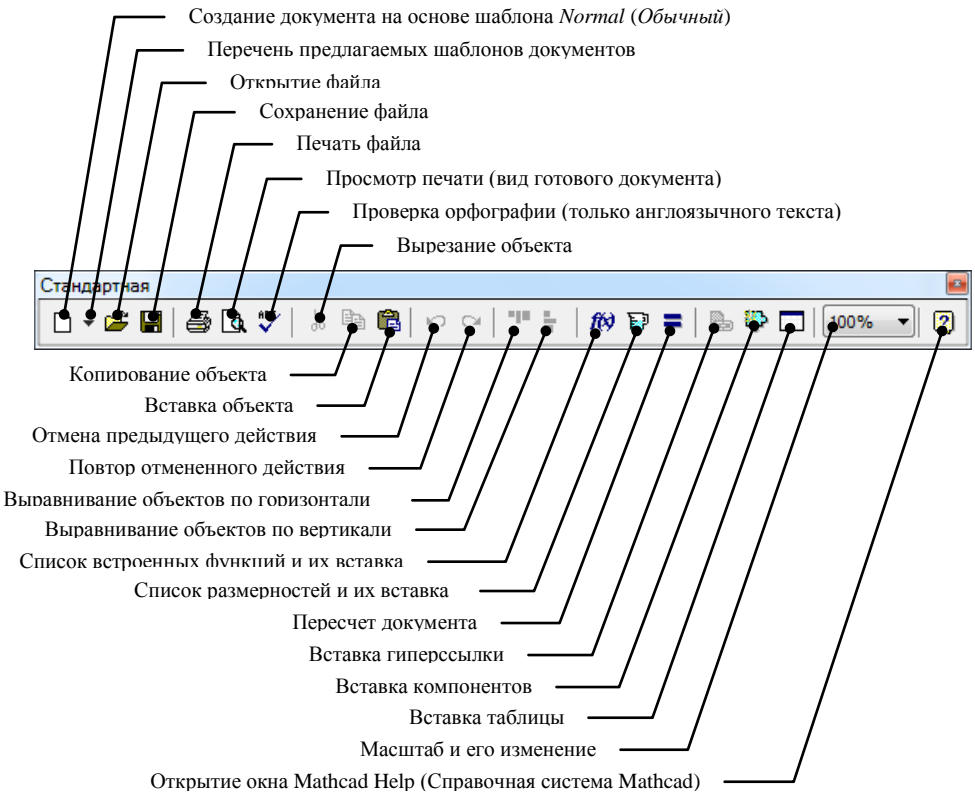


Рисунок 1.3 – Панель «Стандартная»

• *Formatting (Форматирование)* предназначена для форматирования текста и формул.

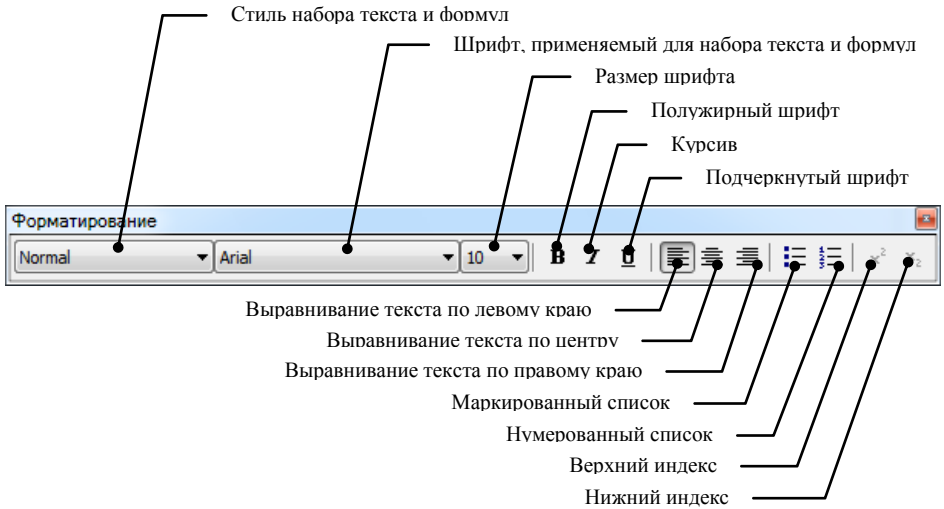


Рисунок 1.4 – Панель «Форматирование»

Math (Математика) служит для вставки математических символов и операторов в документы. Щелчок мышью на любом из значков вызывает вставку на место курсора в рабочем документе соответствующего этому значку символа или шаблона выполнения математической операции.

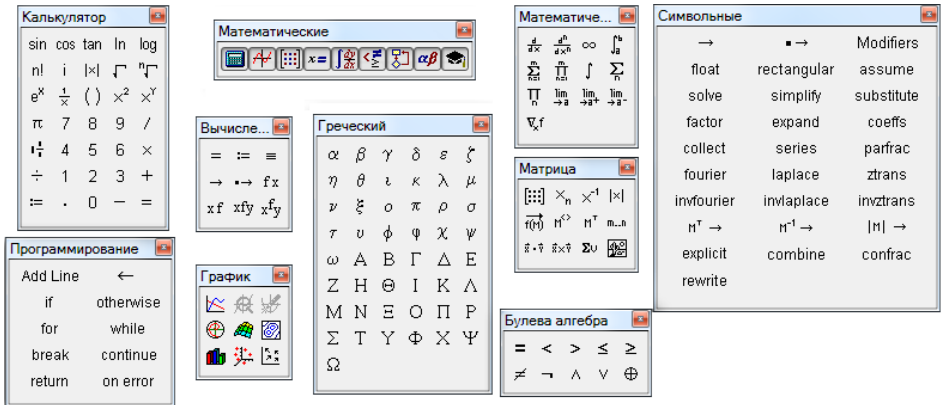


Рисунок 1.5 – Математическая панель и её подчиненные панели

В состав панели входят:

- ✓ *Calculator (Калькулятор)* – вставка шаблонов основных математических операций, цифр, знаков арифметических операций.
 - ✓ *Graph (График)* – вставка шаблонов графиков.
 - ✓ *Matrix (Вектор и матрица)* – вставка шаблонов векторов и матриц а также матричных операций.
 - ✓ *Evaluation (Вычисления)* – операторы присвоения значений и вывода результатов расчета.
 - ✓ *Calculus (Математический анализ)* – вставка шаблонов дифференцирования, интегрирования, суммирования.
 - ✓ *Boolean (Булева алгебра)* – вставка логических (булевых) операторов.
 - ✓ *Programming (Программирование)* – операторы, необходимые для создания программных модулей.
 - ✓ *Greek (Греческие символы)* – вставка греческих символов.
 - ✓ *Symbolic (Символьные преобразования с ключевыми словами)* – вставка операторов символьных вычислений
 - *Resources (Дополнительные ресурсы)* содержит список электронных книг, включенных в оболочку Mathcad.
 - *Controls (Контроль)* содержит кнопки для дополнительного контроля работы Mathcad-документа.
 - *Debug (Отладка)* появилась в Mathcad 13, служит для трассировки выполнения программ.
 - Панель ресурсов
- В выпадающем меню панели Resources (см. рисунок 1.6) содержится список электронных книг, включенных в пакет Mathcad:

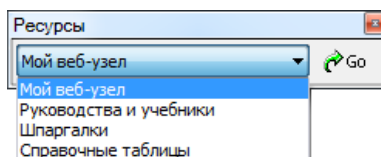


Рисунок 1.6 – Математическая панель и её подчиненные панели

2 Ввод, редактирование и форматирование математических выражений

Перед началом работы курсор на экране имеет вид крестика. В момент ввода выражения курсор приобретает вид синего уголка, обрамляющего вводимое выражение. В рабочем документе нужно вводить какие-либо буквенные выражения и присваивать им численные значения. Имя выражения может состоять из латинских, русских, греческих и других букв и цифр, знаков подчеркивания [_], штриха [‘], символа процента [%], вводимых с клавиатуры.

Имена переменных и функций не могут начинаться с цифры, знака подчеркивания, штриха, символа процента, не могут включать в себя пробелы. Символ бесконечности может быть только первым символом в имени.



Mathcad воспринимает прописные и строчные буквы как различные идентификаторы, то же касается букв, изображенных различными шрифтами, – это разные имена.

Mathcad не делает различий между именами переменных и функций. Если вначале определить функцию $f(x)$, а затем – переменную f , окажется невозможно использовать $f(x)$ в расчетах где-либо после определения f .



Mathcad 12 не допускает переопределения значений функций. Mathcad 13-15 его разрешают, но предупреждают о нем появлением волнистой линии под именем функции. В окне *Preference (Параметры)*, выбрав вкладку *Warnings (Предупреждения)*, можно такие предупреждения отключить.



Некоторые имена уже используются Mathcad для встроенных констант, единиц измерения и функций. Имена можно переопределить, но следует учитывать, что это уничтожит их встроенные значения и этими константами и функциями пользоваться будет нельзя.

2.1 Курсоры Mathcad

Визир – + (крестообразный курсор) – используется для размещения новых выражений, графиков и текстовых областей. Визир может появляться только в свободном месте документа. При начале печати курсор принимает другую форму. Для его перемещения необходимо щелкать в свободном месте документа или используйте клавиши перемещения ([←],[→],[↑],[↓]). Для перемещения ви-

зира на четверть высоты окна вверх используются сочетания клавиш **[PageUp]** и **[PageDn]**. Чтобы перемещать визир вверх или вниз на высоту окна необходимо использовать сочетания клавиш **[Ctrl]+[PageUp]** и **[Ctrl]+[PageDn]**.

Маркер ввода – **|** – используется в выражениях для вставки и удаления отдельных символов, скобок и операторов. Может использоваться в области любого типа. В тексте для его перемещения можно использовать все четыре клавиши перемещения – **[←]**, **[→]**, **[↑]**, **[↓]** – или щелчком мыши перемещаться в другую точку ввода текста. В выражении можно использовать клавиши **[←]** или **[→]** или щёлкнуть в имени или числе.

Выделяющая рамка – **┌** – используется в выражениях для вставки и удаления операторов, чисел, а также имен функций и переменных. Может использоваться только в математической и графической областях. Угол указывает направление вставки или удаления. Для изменения направления необходимо нажать клавишу **[Ins]**.

2.2 Определение значений переменных

Для того, чтобы можно было вычислить выражение, зависящее от каких-либо переменных, значения этих переменных должны быть определены. Для этого нужно:

1. Ввести имя переменной.



2. Ввести двоеточие **[:]**, что приведёт к появлению знака присваивания (**[:=]**) и следующего за ним поля ввода.



3. Напечатать в поле ввода число или выражение. Mathcad вычислит соответствующее значение и присвоит его имени переменной.



Если во вводимом выражении присутствуют другие переменные, они должны быть определены заранее. В противном случае переменные, значения которых не определены к моменту вычисления выражения, будут отмечены на экране дисплея выделением красным цветом.

Для создания:

• дискретных аргументов – ввести начальное значение, затем точку с запятой [;] для создания знака диапазона и после конечное значение аргумента. При этом шаг приращений по умолчанию будет равен 1. Если задать три значения аргумента, то шаг будет равен разнице между вторым и первым значением аргумента;


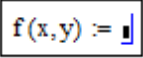
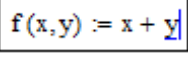
$x := 1, 3..20$

$x := 0..5$

- степени – нажать клавишу [^]; x^y
- индексов переменной - ввести точку [.]; y_1
- индексов массивов – нажать клавишу левой скобки [[]; M_{ij}
- матриц, интегралов и т.п. – вызвать необходимую палитру, задать размерности и ввести в поля ввода соответствующую число или выражение.

2.3 Определение функции

Функция есть правило, согласно которому проводятся некоторые вычисления с её аргументами и вырабатывается её числовое значение. Для того, чтобы определить функцию, нужно:

1. Ввести имя функции и левую скобку [(].
2. Ввести список аргументов, отделяемых друг от друга запятыми и закончить его правой скобкой [)]. 
3. Ввести двоеточие [:], что приведет к появлению знака присваивания (:=) и следующего за ним поля ввода. 
4. Напечатать в поле ввода выражение. 

2.4 Вычисление выражения

Для того, чтобы получить числовой результат, нужно:

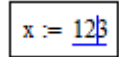
1. Ввести выражение, значение которого нужно вычислить. $2 + 3^2$
2. Ввести знак равенства [=]. $2 + 3^2 = 11$

2.5 Редактирование формул

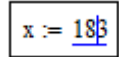
Замена символа:

Если необходимо исправить несколько символов:

1. Щёлкнуть на имени или числе. Если необходимо, нажать клавишу [↓], чтобы превратить выделяющую рамку в маркер ввода. Для перемещения маркера использовать клавиши [←] и [→].

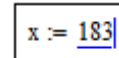


2. Напечатать буквы или цифры. Чтобы удалить символ слева от маркера, нажать [**Backspace**].



Другой способ:

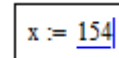
1. Щёлкнуть на имени или числе. Если необходимо, нажать клавишу [↑], чтобы превратить маркер ввода в выделяющую рамку и заключить в неё имя или число.



2. Напечатать буквы или цифры. Чтобы удалить символ слева от маркера, нажать [**Backspace**].



3. Щёлкнуть в поле ввода и напечатать новое имя или число.

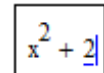


Вставка оператора:

1. Щёлкнуть на операнде, чтобы заключить его в выделяющую рамку. Если нужно увеличить выделяющую рамку следует использовать клавишу [↑].



2. Набрать комбинацию клавиш (сумма, разница, умножить, разделить), задающую оператор.



4. Чтобы вставить оператор перед выделенным выражением нужно нажать клавишу [**Ins**] прежде, чем начинать печатать.



Замена оператора:

1. Выделить операнд перед/после заменяемого оператора.



2. Нажать [**Backspace**]/[**Del**], чтобы удалить оператор.



3. Напечатать новый оператор.

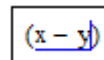


Вставка и удаление скобок:

1. Заключить выражение в выделяющую рамку.



2. Нажать апостроф ['] чтобы вставить скобки.



3. Нажать [Del], чтобы удалить пару скобок.



Вставка и удаление знака:

1. Щёлкнуть на выражении, чтобы заключить его в выделяющую рамку. Использовать [↑], если нужно увеличить выделяющую рамку.



2. Для вставки знака нажать [Ins].



3. Нажать знак минус.

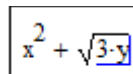


4. Для удаления знака заключить выражение целиком, включая знак минус, в выделяющую рамку. Далее нажать [Backspace].



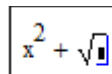
Перемещение частей формулы:

1. Заключить выражение, которое следует переместить, в выделяющую рамку.



$$y := \sin(\bullet)$$

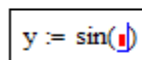
2. Выполнить команду *Вырезать* или *Копировать* из меню *Правка*, чтобы вырезать или скопировать выражение в *Буфер Обмена*.



$$y := \sin(\bullet)$$

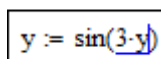
3. Щёлкнуть в свободном пространстве или в поле ввода, куда необходимо вставить выражение.

$$x^2 + \sqrt{\bullet}$$



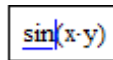
4. Выполнить команду *Вставить* из меню *Правка*, чтобы вставить выражение.

$$x^2 + \sqrt{\bullet}$$



Применение функции к выражению

1. Заключить выражение в выделяющую рамку.
2. Нажать апостроф ['].
3. Нажать [**Пробел**] чтобы заключить скобки в рамку.
4. Нажать [**Ins**].
5. Напечатать имя функции.



2.6 Ввод и редактирование текста

Создание текстовых областей:

1. Щёлкнуть в месте создания области.
2. Выбрать команду *Регион текста* из меню *Вставка*. Появится текстовая рамка.
3. По мере ввода текста текстовая рамка будет увеличиваться. Она исчезнет, как только если щёлкнуть вне текстовой области.



4. Чтобы покинуть текстовую область, необходимо щёлкнуть вне её. Не стоит нажимать [**Enter**], это приведёт к переходу на новую строку внутри текста.



Данную операцию можно произвести вводом с клавиатуры символа (").
При этом на экране появляется текстовая область, в которой вводится текст.

Замена символа:

1. Щёлкнуть на имени или числе. Для перемещения маркера необходимо использовать клавиши [**←**] и [**→**].
2. Напечатать буквы или цифры. Чтобы удалить символ слева от маркера, нужно нажать [**Backspace**].

Скопировать или вырезать текст:

1. Выделить область пунктирным выделяющим прямоугольником.
2. Выполнить команду *Копировать* из меню *Правка*, чтобы скопировать текст в *Буфер Обмена* или выполнить команду *Вырезать*, чтобы вырезать.

Вставить текст:

1. Поместить курсор в то место, куда нужно вставить текст.
2. Выполнить команду *Вставить* из меню *Правка*.

Перемещение текста:

2. Выделить область пунктирным выделяющим прямоугольником.
3. Поместить указатель мыши внутрь прямоугольника.
4. Нажав и удерживая левую кнопку мыши, переместить мышь. Прямоугольник переместится вслед за ней.

Изменение стиля текстовой области:

Чтобы настроить стиль текстовой области необходимо:

1. В главном меню выберите команду *Style* → *Normal* → *Modify* → *Font* (*Стиль* → *Обычный* → *Изменить* → *Шрифт*).
2. В появившемся диалоговом окне выберите нужные шрифт (*Font*), форму шрифта (*Style Font*) и размер (*Size*).

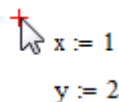


Шрифты Mathcad плохо воспринимают кириллицу. В частности, привычный по работе в Word шрифт Times New Roman кириллицу не принимает. С кириллицей работают шрифты System и Ms Sans Serif.

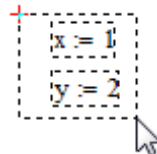
2.7 Операции с областями

Выделение областей:

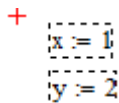
1. Щёлкнуть мышью рядом с областями, которые нужно выделить. Это закрепит один угол выделяющего прямоугольника.



2. Нажав и удерживая левую кнопку мыши, переместить мышь. Пунктирный выделяющий прямоугольник возникнет из закреплённой точки.



3. Когда выделяющий прямоугольник достигнет всех требуемых областей, отпустить кнопку мыши. Каждая выделенная область окажется охваченной пунктирным выделяющим прямоугольником.



Перемещение выделенных областей:

1. Поместить указатель мыши над одной из областей, выделенной пунктирным прямоугольником. Указатель мыши изменит форму со стрелки на черную ладонь.



2. Нажав и удерживая левую кнопку мыши, переместить мышь. Контур области последуют за движением мыши.

3. Когда контуры достигнут желаемого расположения, отпустить кнопку мыши.

Копирование областей:

1. Выделить области.

2. Выбрать команду *Копировать* из меню *Правка*, чтобы скопировать выделенные области в *Буфер Обмена*.

3. Щелкнуть мышью там, куда нужно поместить копию областей. Убедится, что выбрано свободное место – курсор должен иметь форму визира.

4. Выбрать команду *Вставить* из меню *Правка*.



Для выполнения данных операций также возможно использование следующих комбинаций на клавиатуре: **[Ctrl]+[C]** (для копирования) → **[Ctrl]+[V]** (для вставки). Данные сочетания также можно найти слева от команд в пункте меню *Правка*.

Удаление областей:

1. Выделить области.

2. Выбрать команду *Удалить* из меню *Правка*.

Выравнивание областей:

1. Выделить области.

2. Выбрать команду *Выровнять Области* из меню *Правка*.

3. Задать направление выравнивания – *Горизонтально* или *Вертикаль-*

но.



Выполнение данных операция может привести к перекрытию областей.

При работе с Mathcad рекомендуется постоянно пользоваться правой кнопкой мыши. В возникающем при этом контекстном меню появляются, как правило, самые нужные в данный момент пункты, в частности, почти всегда присутствуют пункты *Cut (Вырезать)*, *Copy (Копировать)*, *Paste (Вставить)*.

2.8 Числовые константы. Ввод греческих букв

Введенная с клавиатуры латинская буква e внутри математического выражения означает основание натурального логарифма, $e=2,718$. Это значение можно отменить и присвоить e другое значение, используя знак локального присваивания ($:=$), например: $e:=2$.

Знак бесконечности ∞ можно вставить с математической панели *Математический анализ*.

Часто используемое в выражениях число π можно набрать с математической панели Calculator (Калькулятор), где есть кнопка π , или нажать аналогичную кнопку панели греческих букв.



Есть и более простой способ ввода греческих букв. Необходимо с клавиатуры набрать латинский аналог греческой буквы, как правило, первую букву в латинском названии греческой буквы. Введя аналог греческой буквы, следует нажать комбинацию клавиш **[Ctrl]+[G]**. На экране появится греческая буква – прописная или строчная в зависимости от того, в верхнем или в нижнем регистре был набран ее латинский аналог.

| | | | | | | | | | | | | | | | | | | | | |
|-------------------|----------|---------|--------|----------|------------|--------|----------|-----------|-------|-------|----------|--------|-------|--------|--------|----------|--------|----------|-------|---------|
| Греческие буквы | α | β | χ | δ | ϵ | η | γ | λ | μ | ν | ω | ϕ | π | ψ | ρ | σ | τ | θ | ξ | ζ |
| Латинские аналоги | a | b | c | d | e | h | g | l | m | n | w | f | p | y | r | s | t | q | x | z |

Если какой-либо константе или переменной не присвоено никакого значения левее и выше ее положения на экране, то она окрашена в красный цвет, что свидетельствует об ошибке.

2.9 Функции пользователя

Удобство и эффективность расчетов в Mathcad определяется прежде всего возможностью и легкостью создания функций пользователя. Функция пользователя не вычисляется Mathcad, а принимается к сведению. Для вычисления функции надо задать численные значения всех параметров в имени слева название функции (с параметрами в скобках), справа, после оператора присваивания ($:=$), вычисляемое выражение.

Переменные величины, входящие в правую часть, должны быть записаны в параметры после имени функции. Все величины из правой части, не входя-

щие в параметры левой части, должны быть заданы численно левее и выше функции пользователя. В противном случае Mathcad указывает на ошибку, окрашивая не заданную величину в красный цвет. При выделении функции щелчком мыши появляется текст сообщения об ошибке This variable is not defined above (Эта переменная не определена ранее).

2.10 Дискретные переменные. Построение таблиц

Выражение «дискретная переменная» – диапазон изменения переменной: вместо непрерывной переменной используется ряд чисел, выстроенных в порядке возрастания или убывания. Дискретная переменная задает ряд значений переменной, для которых вычисляется функция пользователя. Этот ряд значений функции можно вывести в виде графика или таблицы.

Определение дискретной переменной имеет вид $x := 0..5$, что означает задание ряда значений $x = 0, 1, 2, 3, 4, 5$.

Дискретная переменная может задавать как целые, так и дробные значения переменной, но обязательно равноотстоящие друг от друга, например:

- $x := 0..5$ – ряд целых чисел от 0 до 5;
- $x := 1,1.1..5$ – ряд дробных чисел, где 1 – первое число, 1.1 – второе число, 5 – последнее число. Интервал между числами $1.1-1=0.1$;
- $x := \min, \min + \Delta.. \max$ – ряд чисел, где \min – первое, Δ – шаг (интервал), \max – последнее число (\min , Δ и \max должны быть заданы заранее). Такая форма записи удобна, когда рассматриваются разные варианты одного расчета и изменение констант позволяет мгновенно пересчитать результаты и перестроить графики.



Двоеточие – знак диапазона нельзя набирать с клавиатуры, нажимая два раза клавишу «точка». Надо на клавиатуре нажать клавишу [;] (точка с запятой) или в математической панели выбрать *Matrix* (значок матрицы) $\rightarrow m..n$.

Если константы, входящие в правую часть функции пользователя, не задаются непосредственно перед использованием функции, Mathcad берет их значения, использовавшиеся в последний раз перед данным расчетом. Необходимо проверить эти значения. Следует набрать имя параметра и нажать клавишу (=).

2.9 Форматирование чисел

В Mathcad на результат расчета повлиять нельзя, но можно изменить формат вывода чисел. Mathcad вычисляет все выражения с точностью 20 знаков, но выводит на экран не все значащие цифры.

Установив указатель мыши на нужном численном результате расчета, нужно сделать двойной щелчок левой кнопкой мыши. Откроется окно форматирования чисел *Result Format (Формат результата)*, открытое на пункте *Number Format (Формат чисел)*. В этом окне можно выбрать следующие форматы:

- *General (Общий)* – принят по умолчанию. Числа отображаются с порядком. Количество знаков перед запятой определяется в пункте *Exponential threshold (Порог экспоненты)*.

- *Scientific (Научный)* – числа отображаются только с порядком: $1,22 \cdot 10^5$.

- *Decimal (Десятичный)* – десятичное представление чисел с плавающей запятой: 12,2564.

- *Engineering (Инженерный)* – числа отображаются только с порядком, кратным 3: $1,22 \cdot 10^6$.

- *Fraction (Дробь)* – в виде правильной или неправильной дроби: $\frac{2}{3} / 1\frac{2}{3}$.

В дробном формате можно выбрать уровень точности (*Level of accuracy*) и смешанные числа (*Use fixed number*).

Выбранные установки могут быть применены только к выделенному числу (если выбрать пункт *OK*) или по умолчанию ко всем числам данного документа (если выбрать пункт *Set as default (Применить по умолчанию)*).

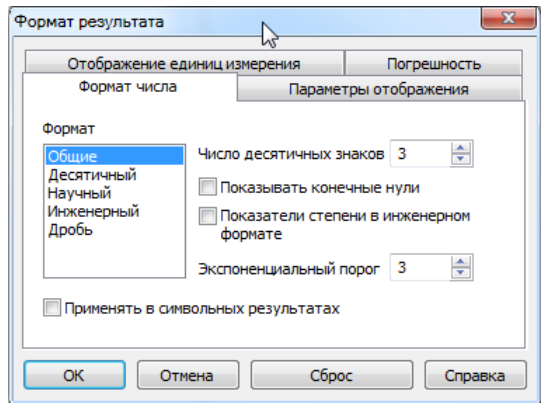


Рисунок 1.3 – Диалоговое окно «Формат результата»

3 Основные арифметические операции.

1. Ввод пары чисел:

$$a := 23.4 \quad b := 78$$

2. Сумма и разность:

$$a + b = 101.4 \quad a - b = -54.6$$

3. Произведение и отношение:

$$a \cdot b = 1.825 \cdot 10^3 \quad \frac{a}{b} = 0.3$$

4. Квадратный корень:

$$\sqrt{a} = 4.837$$

$$\sqrt{b} = 8.832$$

5. Вычисление корня n-ой степени:

$$n := 3$$

$$\frac{1}{a^n} = 2.86$$

$$\frac{1}{b^n} = 4.273$$

6. Степень:

$$a^b = 6.293 \cdot 10^{106}$$

3.1 Тригонометрические функции

1. Вычисление значений в радианах.

1.1 Значение аргумента:

$$x := 3.34$$

1.2 Соответствующие значения тригонометрических функций:

$$\sin(x) = -0.197 \quad \cos(x) = -0.98$$

$$\sec(x) = -1.02 \quad \csc(x) = -5.073$$

$$\tan(x) = 0.201 \quad \cot(x) = 4.974$$

2. Вычисление значений в градусах.

2.1 Значение аргумента:

$$x := -300 \quad x := x \text{ deg}$$

2.2 Соответствующие значения тригонометрических функций:

$$\sin(x) = 0.866 \quad \cos(x) = 0.5$$

$$\sec(x) = 2 \quad \csc(x) = 1.155$$

$$\tan(x) = 1.732 \quad \cot(x) = 0.577$$

3.2 Логарифмы

1. Натуральные и по основанию 10.

Вычисляет натуральные и по основанию 10 логарифмы от положительных вещественных чисел.

1.1 Ввод положительного числа:

$$x := 5.67$$

1.2 Вычисление натурального логарифма:

$$\ln(x) = 1.735$$

1.3 Вычисление логарифма по основанию 10:

$$\log(x) = 0.754$$

2. Логарифмы по произвольному основанию

Вычисляет логарифм положительного вещественного числа по произвольному основанию.

2.1 Ввод положительного числа:

$$x := 12.71$$

2.2 Ввод основания:

$$b := 2$$

2.3 Вычисление логарифма по данному основанию:

$$\log_b(b, x) := \frac{\ln(x)}{\ln(b)}$$

$$\log_b(b, x) = 3.676$$

4 Символьные операции

4.1 Упрощение выражений

Команда *Simplify* (*Упростить*) служит для основных алгебраических и тригонометрических упрощений выбранного выражения. Она выполняет арифметические преобразования, сокращает общие множители, использует основные тождества для тригонометрических и обратных функций, упрощает степени. Можно упрощать как все выражение целиком, так и его части. Если преобразование всего выражения не дает желаемого результата, нужно попытаться выделить и упростить часть выражения.

Пример:

Упростить выражение:

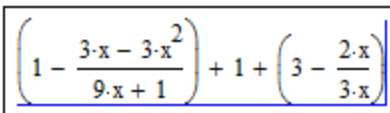
$$\left(1 - \frac{3x - 3x^2}{9x + 1}\right) + 1 + \left(3 - \frac{2x}{3x}\right)$$

Порядок выполнения работы:

1. Ввести выражение, где при вводе дробей или скобок необходимо соблюдать размещение синего уголка, т.к. при неправильном вводе результат будет ошибочным или вообще не будет вычисления.

$$\left(1 - \frac{3 \cdot x - 3 \cdot x^2}{9x + 1}\right) + 1 + \left(3 - \frac{2x}{3x}\right)$$

2. Выделить выражение или разместить синий уголок так, чтобы он охватывал все выражение.


$$\left(1 - \frac{3 \cdot x - 3 \cdot x^2}{9 \cdot x + 1}\right) + 1 + \left(3 - \frac{2 \cdot x}{3 \cdot x}\right)$$

3. В меню *Символьные операции* выбрать команду *Упростить*.

4. Полученный результат должен появиться снизу от выражения.

$$\frac{13}{3} - \frac{3 \cdot x - 3 \cdot x^2}{9 \cdot x + 1}$$

4.2 Раскрытие выражений

Операция символьного разложения по смыслу противоположна операции упрощения. В ходе разложения раскрываются все суммы и произведения, а сложные тригонометрические выражения разлагаются с помощью тригонометрических тождеств. Если выражение – дробь, числитель будет разложен и выражение будет представлено как сумма дробей.

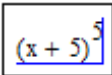
Пример:

Раскрыть скобки:

$$(x + 5)^5$$

Порядок выполнения работы:

1. Ввести и выделить выражение.



The image shows a screenshot of a software interface where the expression $(x + 5)^5$ is highlighted with a blue rectangular box. The text is in a blue font.

2. В меню *Символьные операции* выбрать команду *Развернуть*. Полученный результат должен появиться снизу от выражения.

$$x^5 + 25x^4 + 250x^3 + 1250x^2 + 3125x + 3125$$

4.3 Разложение на множители

Команда *Factor* (*Разложить на множители*) позволяет разложить полиномы на произведение более простых полиномов, а целые числа – на простые множители. Команда будет объединять сумму дробей в одну дробь и упрощать многоэтажную дробь с несколькими дробными чертами.

Пример:

Разложить на множители:

$$\frac{2x^3 - 3x + 1}{x^3 + 2x^2 - 9x - 18}$$

Порядок выполнения работы:

1. Ввести и выделить выражение.

$$\frac{2x^3 - 3x + 1}{x^3 + 2x^2 - 9x - 18}$$

2. В меню *Символьные операции* выбрать команду *Факторизировать*.

Полученный результат должен появиться снизу от выражения.

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

Пример:

Разложить на множители число 124678488

Порядок выполнения работы:

1. Ввести и выделить выражение.

$$124678488$$

2. В меню *Символьные операции* выбрать команду *Факторизировать*.

Полученный результат должен появиться снизу от выражения.

$$2^3 \cdot 3 \cdot 11 \cdot 347 \cdot 1361$$

4.4 Приведение подобных слагаемых

Команда *Collect (Привести подобные слагаемые)* объединяет члены, содержащие одинаковые степени выделенной курсором переменной.

Пример:

Привести подобные слагаемые:

$$x^2 + 2x + 3x^3 + 8x^2 + 6x + 15x^2 + 50 + 10x^3$$

Порядок выполнения работы:

1. Ввести выражение и выделить переменную.

$$x^2 + 2x + 3x^3 + 8x^2 + 6x + 15x^2 + 50 + 10x^3$$

2. В меню *Символьные операции* выбрать команду *Сборка*. Полученный результат должен появиться снизу от выражения.

$$13x^3 + 24x^2 + 8x + 50$$

4.5 Определение коэффициентов полинома

Многие выражения или части выражения могут быть представлены в виде полиномов от выделенной переменной. Коэффициенты полинома могут быть найдены символьным процессором. При этом коэффициенты могут быть сложными функциями других переменных.

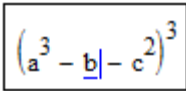
Пример:

Найти коэффициенты полинома по переменной b :

$$(a^3 - b - c^2)^3$$

Порядок выполнения работы:

1. Ввести выражение и выделить переменную.


$$(a^3 - \underline{b} - c^2)^3$$

2. В меню *Символьные операции* выбрать команду *Полиномиальные коэффициенты*. Полученный результат должен появиться снизу от выражения.

$$\begin{pmatrix} a^9 - 3 \cdot a^6 \cdot c^2 + 3 \cdot a^3 \cdot c^4 - c^6 \\ 6 \cdot a^3 \cdot c^2 - 3 \cdot a^6 - 3 \cdot c^4 \\ 3 \cdot a^3 - 3 \cdot c^2 \\ -1 \end{pmatrix}$$

4.6 Вычисление пределов

Для вычисления пределов в Mathcad необходимо выполнить следующие действия:

1. На математической панели выбрать кнопку со знаком интеграла, откроется панель *Calculus (Вычисление)*, на которой внизу есть три оператора вычисления пределов. Необходимо выбрать один из них.
2. Ввести выражение в поле ввода справа от *lim*.
3. В поле ввода под словом *lim* ввести имя переменной, по которой надо вычислить предел, и ее предельное значение.
4. Выделить уголком или черным цветом все выражение целиком.

5. В главном меню Mathcad выбрать *Symbolics* → *Evaluate* → *Symbolically* (*Символьные вычисления* → *Вычислить* → *Аналитически*). Mathcad возвращает значение предела, если оно существует.

Вычислить предел выражения можно только символично.

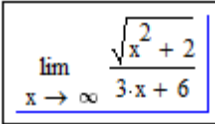
Пример:

Вычислить предел:

$$\lim_{x \rightarrow \infty} \frac{\sqrt{x^2 + 2}}{3x + 6}$$

Порядок выполнения работы:

3. Ввести и выделить выражение.



4. В меню *Символьные операции* выбрать команду *Вычислить* → *Аналитически*. Полученный результат должен появиться снизу от выражения.

$$\frac{1}{3}$$

4.7 Вычисление производной или интеграла

С помощью меню *Symbolics* производную или интеграл можно вычислить двумя способами:

1. В нужном выражении, записанном без знаков производной или интеграла, выделить переменную, по которой надо вычислить производную или взять интеграл. В главном меню выберите команду *Symbolics* → *Variable* → *Differentiate* (*Символьные вычисления* → *Переменная* → *Дифференцировать*) или *Symbolics* → *Variable* → *Integrate* (*Символьные вычисления* → *Переменная* → *Интегрировать*).

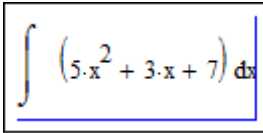
2. На панели *Calculus* (*Вычисления*) выберите знак интеграла или производной и запишите с его помощью нужное выражение. В главном меню выберите команду *Symbolics* → *Evaluate* → *Symbolically* (*Символьные вычисления* → *Вычислить* → *Аналитически*).

Пример:

Вычислить интеграл $\int (5x^2 + 3x + 7)dx$

Порядок выполнения работы:

1. Ввести и выделить выражение.


$$\int (5x^2 + 3x + 7) dx$$

2. В панели *Вычисление* воспользоваться знаком символического равенства.

Полученный результат должен появиться справа от выражения.

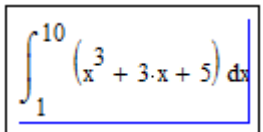
$$\int (5x^2 + 3x + 7) dx \rightarrow \frac{5 \cdot x^3}{3} + \frac{3 \cdot x^2}{2} + 7 \cdot x$$

Пример:

Вычислить интеграл $\int_1^{10} (x^3 + 3x + 5)dx$

Порядок выполнения работы:

1. Ввести и выделить выражение.


$$\int_1^{10} (x^3 + 3x + 5) dx$$

2. На панели *Вычисление* воспользоваться знаком символического равенства. Полученный результат должен появиться справа от выражения. Для вывода результата в виде числа следует добавить в конце выражения знак равенства (=).

$$\int_1^{10} (x^3 + 3x + 5) dx \rightarrow \frac{10773}{4} = 2.693 \times 10^3$$

Пример:

Вычислить производную $\left(\sin x + \frac{\cos(x)}{x} \right)'$

Порядок выполнения работы:

1. Ввести и выделить выражение.

$$\frac{d}{dx} \left(\sin(x) + \frac{\cos(x)}{x} \right)$$

2. На панели *Вычисление* воспользоваться знаком символьного равенства. Полученный результат должен появиться справа от выражения.

$$\frac{d}{dx} \left(\sin(x) + \frac{\cos(x)}{x} \right) \rightarrow \cos(x) - \frac{\cos(x)}{x^2} - \frac{\sin(x)}{x}$$

4.8 Разложение на простейшие дроби

Сложное выражение можно преобразовать в сумму простейших дробей:

1. Выделить переменную в заданном выражении.
2. В главном меню выбрать команду *Symbolics* → *Variable* → *Convert to Partial Fraction* (*Символьные операции* → *Переменная* → *Преобразовать к дробно-рациональному виду*).

Mathcad не станет разлагать выражение, которое содержит десятичные точки.

Пример:

Разложить на простейшие дроби $\frac{2x^2 - 3x + 1}{x^3 + 2x^2 - 9x - 18}$

Порядок выполнения работы:

1. Ввести выражение и выделить переменную.

$$\frac{2 \cdot x^2 - 3 \cdot x + 1}{x^3 + 2 \cdot x^2 - 9 \cdot x - 18}$$

5. В меню *Символьные операции* выбрать команду *Символьные операции* → *Переменная* → *Преобразовать к дробно-рациональному виду*. Полученный результат должен появиться снизу от выражения.

$$\frac{1}{3 \cdot (x - 3)} - \frac{3}{x + 2} + \frac{14}{3 \cdot (x + 3)}$$

Векторы и матрицы.

1 Создание массивов

Есть три способа создания массива чисел:

1. Заполнение шаблона матрицы, содержащего пустые места ввода чисел, что подходит для ввода небольших массивов (не более 100 элементов).
2. Использование дискретной переменной. Этот метод подходит, когда есть явная формула для вычисления элементов массива.
3. Считывание данных из файлов.

1.1 Заполнение матрицы по шаблону

1. Установить курсор в то место, где надо создать матрицу.
2. Щелкнуть на кнопке математической панели *Vector and Matrix Toolbar* (*Векторы и матрицы*), а в появившейся панели инструментов щелкнуть на кнопке *Matrix or Vector* (*Матрица или вектор*). Можно в главном меню Mathcad выбрать команду *Insert* → *Matrix* (*Вставка* → *Матрица*) или нажать комбинацию клавиш [Ctrl]+[m].
3. В появившемся диалоговом окне вписать число строк (*Rows*) и число столбцов (*Columns*), а затем щелкнуть на кнопке *OK*. На месте курсора появится шаблон матрицы.

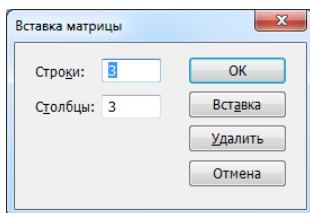


Рисунок 2.1 – Диалоговое окно «Вставка матрицы»

4. В каждое место ввода вписать число, буквенную константу или функцию. Переход от одного места ввода к другому осуществляется клавишами со стрелками. Можно также щелкнуть мышью на нужном месте ввода, но это не так удобно, как перемещение с помощью клавиш.

2.2 Изменение размеров матрицы

Можно изменять размер матрицы, вставляя и удаляя строки и столбцы:

1. Щелкнуть мышью на крайнем элементе матрицы. Синий уголок курсора должен находиться справа от выделенного элемента.
2. Выбрать значок матрицы в математическом меню или нажать комбинацию клавиш **[Ctrl]+[m]**. В появившемся окне вписать число строк и столбцов, которые необходимо вставить или удалить.
3. Щелкнуть на кнопке *Insert (Вставить)* или *Delete (Удалить)*, затем на *OK*. Если необходимо вставить или удалить одну строку, то надо вписать количество строк – 1, количество столбцов – 0 и выбрать пункт *Insert* или *Delete*.



Mathcad вставляет строки или столбцы вправо и вниз от выделенного элемента матрицы, удаляет строки и столбцы вправо и вниз, включая выделенный элемент.

Чтобы удалить всю матрицу, следует выделить ее черным цветом и нажать клавишу **[Delete]** (для безвозвратного удаления) или выбрать *Cut* на стандартной панели Mathcad (чтобы вырезать в буфер). Можно также выбрать команду *Cut* в контекстном меню, появляющемся при щелчке на выделенной матрице правой кнопкой мыши.



При работе с Mathcad рекомендуется часто пользоваться правой кнопкой мыши, проверяя, что можно сделать с помощью контекстного меню. Как правило, в нем появляются самые необходимые в данный момент пункты.

2.3 Нумерация элементов матрицы

Элементы матрицы определяются двумя нижними индексами, элементы вектора – одним.

Для ввода нижнего индекса можно щелкнуть в математической панели на кнопке X_n *Subscript* (Нижний индекс), но лучше использовать клавишу **[[]** (открывающая квадратная скобка), так как во время работы с матрицами вводить нижний индекс приходится очень часто.

Чтобы из матрицы выделить вектор (один из столбцов матрицы), используется верхний индекс. Для выполнения данной операции необходимо:

1. Ввести имя матрицы и выделить его синим уголком курсора.
2. В математическом меню щелкнуть на кнопке *Matrix and Vectors Toolbar* (Панель векторов и матриц) и щелкнуть на кнопке $M^{<0>}$ *Matrix Column* (Столбец матрицы).
3. В появившемся месте ввода вписать номер столбца.

$$M := \begin{pmatrix} 2 & 3 & 1 \\ 4 & 2 & 0 \end{pmatrix} \quad V1 := M^{<0>} \quad V1 = \begin{pmatrix} 2 \\ 4 \end{pmatrix}$$

$$V2 := M^{<1>} \quad V2 = \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

Рисунок 2.2 – Извлечение из матрицы векторов

2.4 Встроенная переменная ORIGIN

Начало нумерации элементов в векторах и матрицах определяется встроенной переменной **ORIGIN**. По умолчанию **ORIGIN**=0, то есть первый элемент вектора, первая строка и первый столбец матрицы имеют индекс ноль.

Чтобы изменить нумерацию индексов в первой строке документа, необходимо набрать (прописными буквами) **ORIGIN**:=1. Можно переопределить встроенную переменную **ORIGIN** другим способом. Для этого необходимо выбрать в главном меню команду *Tools* → *Worksheet Options* (Инструменты → Параметры документа), в открывшемся окне перейти на вкладку *Built-in Variables* (Встроенные переменные) и в поле *Array Origin (ORIGIN)* ввести индекс первого элемента массива. Переменной **ORIGIN** можно присваивать разные значения, в том числе отрицательные.



Исключение составляет функция **Matrix**. При любом значении **ORIGIN** она возвращает матрицу, нумерация элементов в которой начинается с нуля.

В Mathcad есть встроенные функции, в алгоритм которых по умолчанию включены величины с нулевым индексом, например, **mean** (вычисление среднего значения) или **fft** (разложение в ряд Фурье). Если **ORIGIN**:=1 введено с клавиатуры, неизбежна ошибка в расчетах.

2.5 Определение параметров матрицы и векторов

В Mathcad есть следующие встроенные функции для определения параметров матриц и векторов:

Функции, относящиеся к матрицам и векторам:

- **rows(M)** – число строк в матрице или векторе;
- **max(M)** и **min(M)** – максимальное и минимальное значения элементов в матрице или векторе;

Функции, относящиеся только к матрицам:

- **cols(M)** – число столбцов в матрице;
- **tr(M)** – сумма диагональных элементов квадратной матрицы, называемая следом матрицы.

Функции, относящиеся только к векторам:

- **last(M)** – индекс последнего элемента в векторе;
- **length(M)** – число элементов вектора;
- сумма элементов вектора вычисляется нажатием кнопки ΣV на панели

Matrix;

$$M := \begin{pmatrix} 2 & 1 \\ 4 & 3 \\ 5 & 2 \end{pmatrix} \quad n := \text{rows}(M) \quad n = 3$$

$$m := \text{cols}(M) \quad m = 2$$

Рисунок 2.3 – Определение числа строк и столбцов матрицы

$$V := \begin{pmatrix} 4 \\ 1 \\ -7 \end{pmatrix} \quad SV := \Sigma V \quad SV = -2$$

Рисунок 2.4 – Суммирование элементов матрицы

$$\text{ORIGIN} := 1 \quad V := \begin{bmatrix} 4 \\ 3 \\ 1 \\ -5 \end{bmatrix} \quad K := \text{last}(V)$$

$$K = 4$$

Рисунок 2.5 – Определение индекса последнего элемента вектора

$$V := \begin{bmatrix} 4 \\ 3 \\ 1 \\ -5 \end{bmatrix} \quad n := \text{length}(V) \quad n = 4$$

Рисунок 2.6 – Вычисление числа элементов вектора

$$V := \begin{bmatrix} 1 \\ 4 \\ -2 \\ 1 \end{bmatrix} \quad \begin{array}{ll} MV := \max(V) & MV = 4 \\ MV := \min(V) & MV = -2 \end{array}$$

Рисунок 2.7 – Определение минимального и максимального элемента вектора

$$M := \begin{pmatrix} 2 & 4 & -1 \\ 3 & -2 & 7 \\ 8 & -2 & -3 \end{pmatrix} \quad \text{tr}(M) = -3$$

Рисунок 2.8 – Определение суммы элементов на диагонали квадратной матрицы

2.6 Образование новых матриц из уже существующих

Для образования новых матриц из уже существующих существуют следующие функции:

- **augment(A, B)** – объединяет матрицы A и B бок о бок. Матрицы A и B должны иметь одинаковое число строк.

$$M1 := \begin{pmatrix} 1 & 0 & 5 \\ 1 & -4 & 2 \end{pmatrix} \quad M := \text{augment}(M1, M2) \quad M = \begin{pmatrix} 1 & 0 & 5 & 2 & 3 & -1 \\ 1 & -4 & 2 & 4 & -3 & 1 \end{pmatrix}$$

Рисунок 2.9 – Объединение матриц «бок о бок»

- **stack(A, B)** объединяет матрицы друг над другом. Матрицы A и B должны иметь одинаковое число столбцов.

$$\begin{array}{ll} M1 := (1 & 2 & 3) \\ M2 := (4 & 5 & 6) \end{array} \quad M := \text{stack}(M1, M2) \quad M = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

Рисунок 2.10 – Объединение матриц «друг над другом»

- **submatrix(A, irows, jrows, icols, jcols)** – создает матрицу, вырезанную из матрицы A. Новая матрица содержит элементы матрицы A, вырезаемые от ряда irows до ряда jrows, от столбца icols до столбца jcols.

$$M := \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 & 24 \end{pmatrix} \quad M1 := \text{submatrix}(M, 1, 3, 1, 3) \quad M1 = \begin{pmatrix} 7 & 8 & 9 \\ 12 & 13 & 14 \\ 17 & 18 & 19 \end{pmatrix}$$

Рисунок 2.11 – Матрица, полученная «вырезанием» центральной части от предыдущей матрицы

2.7 Ввод больших массивов

В Mathcad есть ограничения на размер вводимых массивов. При использовании шаблона матрицы можно создать массив, состоящий не более чем из 100 элементов. Если надо ввести массив больших размеров, используются встроенные функции **augment** и **stack**.

Можно ввести массив практически неограниченных размеров с использованием дискретной переменной по какой-либо расчетной формуле.

ORIGIN:= 1

n := 5

m := 5

i := 1..n

j := 1..m

$x_i := 0.1i$

$y_j := 0.2j$

$f_{i,j} := \sin(x_i \cdot y_j)$

$$f = \begin{pmatrix} 0.02 & 0.04 & 0.06 & 0.08 & 0.1 \\ 0.04 & 0.08 & 0.12 & 0.159 & 0.199 \\ 0.06 & 0.12 & 0.179 & 0.238 & 0.296 \\ 0.08 & 0.159 & 0.238 & 0.315 & 0.389 \\ 0.1 & 0.199 & 0.296 & 0.389 & 0.479 \end{pmatrix}$$

Рисунок 2.12 – Задание матрицы при помощи расчетной формулы

2.8 Сортировка векторов и матриц

В Mathcad имеется несколько встроенных функций для сортировки элементов массива в порядке возрастания или убывания:

- **sort(V)** – сортировка элементов вектора в порядке возрастания;

$$V := \begin{pmatrix} 5 \\ 2 \\ 4 \\ 1 \\ 3 \end{pmatrix} \quad V := \text{sort}(V) \quad V = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix}$$

Рисунок 2.13 – Сортировка элементов вектора по возрастанию

- **reverse(V)** – перестановка элементов вектора в обратном порядке;

$$V := \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix} \quad V := \text{reverse}(V) \quad V = \begin{pmatrix} 5 \\ 4 \\ 3 \\ 2 \\ 1 \end{pmatrix}$$

Рисунок 2.14 – Перестановка элементов вектора в обратном порядке

- **csort(M, i)** – перестановка строк матрицы M в порядке возрастания элементов i-го столбца;

$$M := \begin{pmatrix} 3 & 4 & 5 & 6 \\ 4 & 5 & 1 & 2 \\ 6 & 8 & 2 & 1 \end{pmatrix} \quad M := \text{csort}(M, 2) \quad M = \begin{pmatrix} 4 & 5 & 1 & 2 \\ 6 & 8 & 2 & 1 \\ 3 & 4 & 5 & 6 \end{pmatrix}$$

Рисунок 2.15 – Сортировка матрицы по элементам столбца

- **rsort(M, i)** – перестановка столбцов матрицы M в порядке возрастания элементов i-й строки.

$$M := \begin{pmatrix} 3 & 4 & 5 & 6 \\ 4 & 5 & 1 & 2 \\ 6 & 8 & 2 & 1 \end{pmatrix} \quad M := \text{rsort}(M, 2) \quad M = \begin{pmatrix} 6 & 5 & 3 & 4 \\ 2 & 1 & 4 & 5 \\ 1 & 2 & 6 & 8 \end{pmatrix}$$

Рисунок 2.16 – Сортировка матрицы по элементам столбца

2.9 Матричные операторы

2.9.1 Транспонирование матрицы

Транспонированием называется операция, в результате которой столбцы исходной матрицы становятся строками, а строки – столбцами.

Для выполнения этой операции необходимо:

1. Набрать имя матрицы.
2. Щелкнуть на кнопке со значком матрицы на математической панели.
3. На панели *Matrix (Матрицы)* щелкнуть мышью на значке соответствующей операции, в данном случае M^T .



При вводе больших векторов из соображений экономии места удобно вводить их в виде строки с последующим транспонированием. Все матричные операторы и матричные функции работают только с векторами (в виде столбца), но не со строками, поэтому строки вначале приходится транспонировать в столбец, а после выполнения нужной операции вновь транспонировать в строку.

$$V := \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix} \quad V^T := V^T \quad V^T = (2 \ 3 \ 1)$$
$$M := \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix} \quad M := M^T \quad M = \begin{pmatrix} 1 & 5 & 9 \\ 2 & 6 & 10 \\ 3 & 7 & 11 \\ 4 & 8 & 12 \end{pmatrix}$$

Рисунок 2.17 – Транспонирование вектора и матрицы

2.9.2 Вычисление определителя матрицы

Для нахождения определителя заданной матрицы на панели *Matrix* необходимо выбрать значок $|x|$, имеющий тройное значение:

$$M := \begin{pmatrix} 2 & 3 \\ 1 & 2 \end{pmatrix} \quad NM := |M| \quad NM = 1$$

Рисунок 2.18 – Вычисление определителя матрицы

2.9.3 Нахождение матрицы, обратной заданной

Для нахождения матрицы, обратной заданной, необходимо выбрать значок X^{-1} на панели Matrix.

Произведение прямой матрицы на обратную есть единичная матрица. Иногда единичная матрица необходима для решения матричных уравнений. Для создания единичной матрицы в Mathcad есть встроенная функция **identity(n)**, где n – размер квадратной матрицы.

$$\begin{array}{l}
 M := \begin{pmatrix} 1 & 3 & 4 & 6 \\ 8 & 5 & 4 & 2 \\ 2 & 1 & 4 & 6 \\ 1 & 2 & 9 & 1 \end{pmatrix} \\
 M := M^{-1} \\
 M = \begin{pmatrix} -0.223 & 0.109 & 0.192 & -0.035 \\ 0.389 & 0.055 & -0.404 & -0.017 \\ -0.068 & -0.021 & 0.053 & 0.127 \\ 0.055 & -0.032 & 0.134 & -0.07 \end{pmatrix} \\
 M1 := \text{identity}(4) \\
 M1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
 \end{array}$$

Рисунок 2.19 – Нахождение обратной и создание единичной матрицы

2.9.4 Сложение, вычитание, умножение и возведение в степень матриц и векторов

Складывать и вычитать можно только матрицы одинакового размера. Сложение и вычитание матриц есть операция нахождения матрицы A, все элементы которой равны попарной сумме всех соответствующих элементов матриц B и C, то есть каждый элемент матрицы равен $c_{ij}=a_{ij} \pm b_{ij}$

$$\begin{array}{l}
 A := \begin{pmatrix} 2 & 8 & 5 \\ 4 & 3 & 8 \\ 7 & 1 & 6 \end{pmatrix} \\
 B := \begin{pmatrix} 7 & 6 & 4 \\ 2 & 5 & 8 \\ 1 & 7 & 7 \end{pmatrix} \\
 A + B = \begin{pmatrix} 9 & 14 & 9 \\ 6 & 8 & 16 \\ 8 & 8 & 13 \end{pmatrix} \\
 A - B = \begin{pmatrix} -5 & 2 & 1 \\ 2 & -2 & 0 \\ 6 & -6 & -1 \end{pmatrix}
 \end{array}$$

Рисунок 2.20 – Сложение и вычитание матриц

Для умножения матриц используется традиционный знак умножения (клавиша со звездочкой). Знак умножения в Mathcad-документе по умолчанию обозначается точкой.

$$A := \begin{pmatrix} 2 & 8 & 5 \\ 4 & 3 & 8 \\ 7 & 1 & 6 \end{pmatrix} \quad B := \begin{pmatrix} 7 & 6 & 4 \\ 2 & 5 & 8 \\ 1 & 7 & 7 \end{pmatrix} \quad A \cdot B = \begin{pmatrix} 35 & 87 & 107 \\ 42 & 95 & 96 \\ 57 & 89 & 78 \end{pmatrix}$$

$$\frac{A}{B} = \begin{pmatrix} 0.306 & -0.796 & 1.449 \\ 0.204 & 1.803 & -1.034 \\ 0.673 & 2.116 & -1.946 \end{pmatrix}$$

Рисунок 2.21 – Сложение и деление и вычитание матриц

Вид значка в документе можно поменять. Для этого необходимо щелкнуть правой кнопкой мыши на выражении, в котором есть знак умножения. В открывшемся контекстном меню найти пункт *View Multiplication as (Вид знака умножения)*. Далее следует щелкнуть мышью на одном из предлагаемых семи пунктов – значок изменится.



Точно так же можно изменить вид знака присвоения значения, но делать этого не рекомендуется, чтобы не перепутать знак присвоения значения и знак логического равенства.

По правилу перемножения матриц, число столбцов первой матрицы должно быть равно числу строк второй матрицы, $A(m \times n) \cdot B(n \times k) = C(m \times k)$ (в скобках указаны размеры перемножаемых матриц).



Полезно запомнить, что произведение строки на столбец дает число, а произведение столбца на строку дает полностью заполненную квадратную матрицу.

Возведение матрицы в степень представляет собой простое перемножение матриц, в котором первый и второй множители равны между собой.

$$A := \begin{pmatrix} 1 & 2 & 1 & 3 \\ 3 & 2 & 1 & 2 \\ 4 & 1 & 3 & 2 \\ 1 & 3 & 4 & 3 \end{pmatrix} \quad A := A^2 \quad A = \begin{pmatrix} 14 & 16 & 18 & 18 \\ 15 & 17 & 16 & 21 \\ 21 & 19 & 22 & 26 \\ 29 & 21 & 28 & 26 \end{pmatrix}$$

Рисунок 2.22 – Возведение матрицы в степень

Умножение вектора на число есть вектор, каждый элемент которого умножен на данное число.

$$\begin{array}{l} V := \begin{pmatrix} 1 \\ 4 \\ 3 \end{pmatrix} \quad V1 := -V \quad V1 = \begin{pmatrix} -1 \\ -4 \\ -3 \end{pmatrix} \\ \\ V := \begin{pmatrix} 0.5 \\ 0.2 \\ 0.7 \end{pmatrix} \quad V := V \cdot 10 \quad V = \begin{pmatrix} 5 \\ 2 \\ 7 \end{pmatrix} \end{array}$$

Рисунок 2.23 – Умножение вектора на число

Скалярное произведение векторов дает число, равное сумме попарных произведений элементов этих векторов, $UV = \sum_{i=1}^n U_i V_i$ где n – число элементов каждого вектора, или $UV = |U||V| \cdot \cos(\theta)$ где θ – угол между векторами.

$$\begin{array}{l} V1 := \begin{pmatrix} 2 \\ 3 \\ 5 \end{pmatrix} \quad V2 := \begin{pmatrix} 3 \\ 2 \\ -1 \end{pmatrix} \quad C := V1 \cdot V2 \quad C = 7 \\ \\ V1 := \begin{pmatrix} 2 \\ 3 \\ 5 \end{pmatrix} \quad V2 := (3 \ 2 \ -1) \quad M := V1 \cdot V2 \quad M = \begin{pmatrix} 6 & 4 & -2 \\ 9 & 6 & -3 \\ 15 & 10 & -5 \end{pmatrix} \end{array}$$

Рисунок 2.24 – Скалярное умножение векторов

Векторное произведение двух векторов, \vec{X} и \vec{Y} , с углом между ними θ равно вектору с модулем $|\vec{X}\vec{Y}| = |\vec{X}||\vec{Y}| \cdot \sin(\theta)$ направленному перпендикулярно плоскости, определяемой векторами \vec{X} и \vec{Y} . Оператор векторного произведения можно ввести только с панели *Matrix* при помощи значка $\vec{X} \times \vec{Y}$.,

$$V1 := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad V2 := \begin{pmatrix} 4 \\ 5 \\ 6 \end{pmatrix} \quad P := V1 \times V2 \quad P = \begin{pmatrix} -3 \\ 6 \\ -3 \end{pmatrix}$$

Рисунок 2.25 – Векторное произведение векторов

Векторное произведение можно вычислить только для векторов с тремя элементами.

2.9.5 Оператор векторизации

В Mathcad существует несколько необычный оператор, называемый оператором векторизации и предназначенный для работы с массивами. Он позволяет выполнить однотипную операцию со всеми элементами массива. Для его применения необходимо:

1. Выделить всё выражение.
2. На панели матриц щелкнуть на кнопке $\overline{f(M)}$. Над выделенным выражением появится знак векторизации;
3. Нажать клавишу [=].

$$M := \begin{pmatrix} \frac{\pi}{8} & \frac{\pi}{4} \\ \frac{\pi}{3} & \frac{\pi}{2} \\ \frac{3\pi}{4} & \pi \\ 2\pi & 3\pi \end{pmatrix} \xrightarrow{\cos(M)} \begin{pmatrix} 0.924 & 0.707 \\ 0.5 & 0 \\ -0.707 & -1 \\ 1 & -1 \end{pmatrix}$$

Рисунок 2.26 – Применение оператора векторизации

Mathcad допускает использование вектора в качестве аргумента функции, и в этом случае знак векторизации не требуется. Например, если V – вектор, $\cos(V) = \text{ответ}$ – без знака векторизации.

$$V := \begin{pmatrix} 0 \\ 30 \\ 45 \\ 60 \\ 90 \\ 180 \end{pmatrix} \quad \cos(V \cdot \text{deg}) = \begin{pmatrix} 1 \\ 0.866 \\ 0.707 \\ 0.5 \\ 0 \\ -1 \end{pmatrix}$$

Рисунок 2.27 – Расчет косинуса элементов вектора без применения оператора векторизации

Решение уравнений

1 Решение квадратного уравнения

Используется для нахождения корней многочлена второй степени вида:

$$f(x) = ax^2 + bx + c$$

Порядок выполнения:

1. Вводятся коэффициенты уравнения.
2. Вводится матрица, состоящая из данных коэффициентов, записанных в обратном порядке. Производится её транспонирование.

3. Для нахождения решения используется функция **polyroots**

Пример:

Решить квадратное уравнение:

$$x^2 - 2x - 8 = 0$$

1. Ввод коэффициентов уравнения.
 $a := 1 \quad b := -2 \quad c := -8$
2. Вводится матрица, состоящая из коэффициентов, записанных в обратном порядке. Производится её транспонирование.

$$v := (c \ b \ a)^T$$

3. Для нахождения решения используется функция **polyroots**.

$$r := \text{polyroots}(v) \quad r = \begin{pmatrix} -2 \\ 4 \end{pmatrix}$$

2 Символьное решение уравнений

Порядок выполнения:

1. Ввести уравнение.
2. Синим уголком курсора выделить переменную, относительно которой нужно решить уравнение.

3. В главном меню выбрать команду *Symbolics* → *Variable* → *Solve* (*Символьные вычисления* → *Переменная* → *Решить*).

Пример:

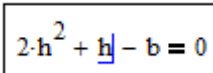
Символьно найти решение уравнения относительно переменной h :

$$2h^3 + h - b = 0$$

1. Ввод уравнения. В качестве знака равно использовать знак Булева равенства (вводится сочетанием клавиш [Ctrl]+[=]).

$$2 \cdot h^2 + h - b = 0$$

2. Выделение переменной, относительно которой проводится решение.



The image shows a rectangular box containing the equation $2 \cdot h^2 + h - b = 0$. The variable h in the second term is highlighted with a blue selection bar, indicating it has been selected for solving.

3. Решение уравнения при помощи команды Symbolics → Variable → Solve (Символьные вычисления → Переменная → Решить).

$$\left(\begin{array}{l} \frac{\sqrt{8 \cdot b + 1}}{4} - \frac{1}{4} \\ -\frac{\sqrt{8 \cdot b + 1}}{4} - \frac{1}{4} \end{array} \right)$$

Недостаток использования меню Symbolics заключается в том, что найденное решение не пересчитывается автоматически при изменении выражения или входящих в него величин и не участвует в последующих расчетах.

Достоинством использования меню Symbolics является то, что ранее принятые численные значения величин не учитываются в символьных расчетах.

Если выделенное выражение не имеет символьного решения (а большинство уравнений не имеет символьного решения), то Mathcad сообщает об ошибке: «No solution was found» («Решение не найдено»).

3 Численное решение уравнений

Функция root решает уравнения итерационным методом секущих и поэтому требует задания перед собой начальных значений. Кроме того, функция root, выполняя вычисления методом спуска, находит и выводит только один корень, ближайший к начальному приближению. Для поиска остальных корней уравнения необходимо задание других начальных значений.

Пример:

Найти корни уравнения в диапазоне от -4 до 3:

$$x^3 - 6x + 2 = 0$$

1. Ввод диапазона, где определяется корень.

$$x := -4..3$$

2. Ввод уравнения.

$$f(x) := x^3 - 6x + 2$$

3. Поиск интервалов, где происходит смена знака функции.

$$x = \quad f(x) =$$

| | |
|----|-----|
| -4 | -38 |
| -3 | -7 |
| -2 | 6 |
| -1 | 7 |
| 0 | 2 |
| 1 | -3 |
| 2 | -2 |
| 3 | 11 |

Данных интервалов 3 (-3...-2, 0...1, 2..3), следовательно уравнение на заданном интервале имеет 3 корня.

4. Задание точности вычисления корня.

$$TOL := 10^{-5}$$

5. Задание начального положения для поиска. Задается как среднее значение между значениями переменной, где происходит смена знака функции.

$$x := \frac{-3 - 2}{2} \quad x = -2.5$$

6. Вычисление корня

$$X1 := \text{root}(f(x), x) \quad X1 = -2.602$$

7. Проверка решения. Значение функции должно быть близко к 0.

$$f(X1) = 3.553 \times 10^{-15}$$

8. Вычисление второго корня.

$$x := \frac{0 + 1}{2} \quad x = 0.5$$

$$X2 := \text{root}(f(x), x) \quad X2 = 0.34$$

$$f(X2) = 6.095 \times 10^{-11}$$

9. Вычисление третьего корня

$$x := \frac{2 + 3}{2} \quad x = 2.5$$

$$X3 := \text{root}(f(x), x) \quad X3 = 2.262$$

$$f(X3) = 3.052 \times 10^{-10}$$

4 Поиск экстремума функции

С помощью функции `root` можно найти и экстремум функции, приравняв производную к нулю. Функции должно предшествовать начальное приближение.

Для нахождения экстремума функции следует:

1. Задать начальное приближение, наиболее близко расположенное к экстремуму. Для его поиска необходимо определить, на каких интервалах происходит смена знака производной функции.

2. Записать выражение с функцией `root`, включив в качестве функции, которая должна быть равна нулю, производную по заданной переменной;

3. Вычислить значение заданной функции от найденного корня.

Пример:

Найти экстремумы уравнения в диапазоне от -4 до 3:

$$x^3 - 6x + 2 = 0$$

1. Ввод диапазона.

$$x := -4..3$$

2. Ввод уравнения.

$$f(x) := x^3 - 6x + 2$$

3. Поиск интервалов, где происходит смена знака производной функции

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

| | |
|----|----|
| -4 | 42 |
| -3 | 21 |
| -2 | 6 |
| -1 | -3 |
| 0 | -6 |
| 1 | -3 |
| 2 | 6 |
| 3 | 21 |

Данных интервалов 2 (-2...-1, 1...2), следовательно уравнение на заданном диапазоне имеет 2 экстремума.

4. Задание точности вычисления экстремума.

$$TOL := 10^{-5}$$

5. Задание начального положения для поиска экстремума. Задается как среднее значение между значениями переменной, где происходит смена знака градиента функции.

$$x := \frac{-2 - 1}{2} \quad x = -1.5$$

6. Нахождение первого экстремума

$$X1 := \text{root}\left(\frac{d}{dx} f(x), x\right) \quad X1 = -1.414$$

7. Вычисление значения функции в экстремуме.

$$f(X1) = 7.657$$

8. Нахождение второго экстремума.

$$x := \frac{1 + 2}{2} \quad x = 1.5$$

$$X2 := \text{root}\left(\frac{d}{dx} f(x), x\right) \quad X2 = 1.414$$

$$f(X2) = -3.657$$

5 Решение систем линейных алгебраических уравнений

Известно, что система линейных уравнений в матричной форме $A \cdot X = B$. Вектор решения получается из $X = A^{-1} \cdot B$.

Для решения систем линейных уравнений в MathCAD существует встроенная функция **Isolve(A,B)**, которая возвращает вектор X для системы линейных уравнений при заданной матрице коэффициентов и векторе свободных членов.

Также используются системы линейных уравнений для вычисления определителя по формулам Крамера – точный метод решения. Решение линейной системы методом Гаусса – приближенный метод.

Функции, предназначенные для решения задач линейной алгебры, можно разделить на три группы:

- функции определения матриц и операции с блоками матриц;
- функции вычисления различных числовых характеристик матриц;
- функций, реализующие численные алгоритмы решения задач линейной алгебры.

Пример:

Решить систему линейных уравнений:

$$\begin{cases} x+2y+3z=1 \\ 2x+3y+z=2 \\ x+2y+5z=3 \end{cases}$$

1. Преобразуем данную систему:

$$\begin{cases} 1x+2y+3z=1 \\ 2x+3y+1z=2 \\ 1x+2y+5z=3 \end{cases}$$

2. Матрица коэффициентов системы:

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 1 & 2 & 5 \end{pmatrix}$$

3. Вектор свободных членов

$$b := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

4. Решение системы

$$X := A^{-1} \cdot b$$

5. Результаты решения

$$X = \begin{pmatrix} 8 \\ -5 \\ 1 \end{pmatrix}$$

6. Проверка решения

$$A \cdot X - b = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

7. Решение с применением функции `Isolve`

$$X := \text{Isolve}(A, b)$$

$$X = \begin{pmatrix} 8 \\ -5 \\ 1 \end{pmatrix}$$

5.1 Решение системы линейных уравнений по формулам Крамера

Порядок выполнения

1. Переменной **ORIGIN** присвоить значение равное единице.
2. Ввести матрицу коэффициентов системы.
3. Ввести вектор свободных членов (столбец правых частей).
4. Вычислить определитель матрицы системы. Система имеет единственное решение, если определитель отличен от нуля.
5. Вычислить определителей матрицы, полученных заменой соответствующего столбца столбцом правых частей.
6. Определить решение системы по формулам Крамера.

Пример:

Решить систему линейных уравнений:

$$\begin{cases} x+2y+3z=1 \\ 2x+3y+z=2 \\ x+2y+5z=3 \end{cases}$$

1. Преобразуем данную систему:

$$\begin{cases} 1x+2y+3z=1 \\ 2x+3y+1z=2 \\ 1x+2y+5z=3 \end{cases}$$

2. Матрица коэффициентов системы:

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 1 & 2 & 5 \end{pmatrix}$$

3. Вектор свободных членов

$$b := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

4. Присваиваем переменной ORIGIN значение 1.

$$\text{ORIGIN} \equiv 1$$

5. Вычисляем определителя матрицы системы.

$$\Delta := |A| \quad \Delta = -2$$

6. Вычисление определителей матрицы

$$\Delta_1 := \begin{vmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 2 & 5 \end{vmatrix} \quad \Delta_2 := \begin{vmatrix} 1 & 1 & 3 \\ 2 & 2 & 1 \\ 1 & 3 & 5 \end{vmatrix} \quad \Delta_3 := \begin{vmatrix} 1 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{vmatrix}$$

$$\Delta_1 = -16$$

$$\Delta_2 = 10$$

$$\Delta_3 = -2$$

7. Определение решения системы по формулам Крамера

$$x := \frac{\Delta_1}{\Delta}$$

$$y := \frac{\Delta_2}{\Delta}$$

$$z := \frac{\Delta_3}{\Delta}$$

$$x = 8$$

$$y = -5$$

$$z = 1$$

5.2 Решение линейной системы методом Гаусса (метод гауссовых исключений)

Порядок выполнения:

1. Переменной ORIGIN присвоить значение равное единице.
2. Ввести матрицу системы и вектор-столбец правых частей.
3. Сформировать расширенную матрицу системы при помощи функции **augment(A,b)**.
4. Привести расширенную матрицу системы к ступенчатому виду при помощи функции **rref(Ar)**.
5. Сформировать столбец решений системы при помощи функции **submatrix(Ag,1,3,4,4)**.

Пример:

Решить систему линейных уравнений:

$$\begin{cases} x+2y+3z=1 \\ 2x+3y+z=2 \\ x+2y+5z=3 \end{cases}$$

1. Преобразуем данную систему:

$$\begin{cases} 1x+2y+3z=1 \\ 2x+3y+1z=2 \\ 1x+2y+5z=3 \end{cases}$$

2. Матрица коэффициентов системы:

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 1 & 2 & 5 \end{pmatrix}$$

3. Вектор свободных членов

$$b := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

4. Присваиваем переменной ORIGIN значение 1.

$$\text{ORIGIN} \equiv 1$$

5. Формирование расширенной матрицы системы

$$\text{Ar} := \text{augment}(A, b) \quad \text{Ar} = \begin{pmatrix} 1 & 2 & 3 & 1 \\ 2 & 3 & 1 & 2 \\ 1 & 2 & 5 & 3 \end{pmatrix}$$

6. Приведение расширенной матрицы системы к ступенчатому виду

$$\text{Ag} := \text{rref}(\text{Ar}) \quad \text{Ag} = \begin{pmatrix} 1 & 0 & 0 & 8 \\ 0 & 1 & 0 & -5 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

7. Формирование столбца решения системы

$$x := \text{submatrix}(\text{Ag}, 1, 3, 4, 4) \quad x = \begin{pmatrix} 8 \\ -5 \\ 1 \end{pmatrix}$$

6 Решения нелинейных уравнений

6.1 Решение трансцендентных уравнений

Многие уравнения, например трансцендентные, не имеют аналитических решений. Однако они могут решаться численными методами с заданной погрешностью. Для простейших уравнений вида $F(x)=0$ решение находится с помощью функции **root**. Эта функция возвращает значение переменной с указанным уровнем, при котором выражение дает 0.

Пример:

Решить уравнение 3-ей степени:

$$x^3 - 6x^2 + 21x - 52 = 0$$

1. Ввод коэффициентов полинома.

$$a3 := 1 \quad a2 := -6 \quad a1 := 21 \quad a0 := -52$$

2. Ввод полинома.

$$f(x) := a3 \cdot x^3 + a2 \cdot x^2 + a1 \cdot x + a0$$

3. Вычисление действительного корня с помощью функции **root**.

$$x := 0 \quad x1 := \text{root}(f(x), x) \quad x1 = 4$$

4. Вычисление двух других корней.

$$i := \sqrt{-1} \quad x := 1 + 1 \cdot i$$

$$x2 := \text{root}\left(\frac{f(x)}{x - x1}, x\right) \quad x2 = 1 + 3.464i$$

$$x3 := \text{root}\left[\frac{f(x)}{(x - x1) \cdot (x - x2)}, x\right] \quad x3 = 1 - 3.464i$$

6.2 Решение систем нелинейных уравнений при помощи блока «Given – Find»

При решении систем нелинейных уравнений используется специальный вычислительный блок, открываемый служебным словом – директивой **Given** – и имеющий следующую структуру:

1. Ввод начальных приближений
2. **Given**
3. Уравнения
4. Выражения с функциями **find** и **minerr**

Пример:

Решить систему нелинейных уравнений:

$$\begin{cases} 2x + y = 5 - z^2 \\ y^3 + 4z = 4 \\ xy + z = e^z \end{cases}$$

1. Задание начальных приближений для всех неизвестных.

$$x := 1 \quad y := 1 \quad z := 0$$

2. Ввод уравнений в блок решения уравнения.

Given

$$2 \cdot x + y = 5 - 2 \cdot z^2$$

$$y^3 + 4 \cdot z = 4$$

$$x \cdot y + z = e^z$$

$$\text{vec} := \text{find}(x, y, z)$$

3. Найденное решение.

$$\text{vec} = \begin{pmatrix} 1.325 \\ 1.823 \\ -0.513 \end{pmatrix}$$

7 Символьное решение систем уравнений

Для того, чтобы найти решение системы уравнений в символьном виде, нужно:

1. Ввести слово **Given**.
2. Ввести уравнения и неравенства ниже слова **Given**.
3. Ввести функцию **Find** с аргументами-неизвестными.
4. Убедиться в том, что команда *Использовать символику* в меню *Математика* отмечена галочкой, в противном случае выберите эту команду.
5. Ввести комбинацию клавиш **[Ctrl]+[+]**. Mathcad выводит стрелку вправо.
6. Щёлкнуть вне области действия функции **Find**. Mathcad выработывает результат в виде вектора. Результаты размещаются в том же порядке, в котором были перечислены неизвестные-аргументы функции **Find**.

Пример:

Найти координаты точки пересечения 2-х прямых:

$$\begin{cases} x+2\pi y=a \\ 4x+y=b \end{cases}$$

1. Ввод уравнений в блок решения уравнения

Given

$$x+2\cdot\pi y=a$$

$$4\cdot x+y=b$$

2. Найденное решение.

$$\text{Find}(x,y) \rightarrow \begin{bmatrix} \frac{1}{(-1+8\cdot\pi)}\cdot(2\cdot\pi b-a) \\ \frac{(4\cdot a-b)}{(-1+8\cdot\pi)} \end{bmatrix}$$

Построение графиков функций и поверхностей

1 Построение двумерного плоского графика

Для построения графика функции необходимо:

1. Ввести уравнение функции
2. Установить крестообразный курсор туда, где надо построить график.
3. На математической панели щелкнуть на кнопке *Graph Toolbar (Панель графиков)*, а на открывшейся панели – на кнопке с изображением плоского графика.
4. В появившемся на месте курсора шаблоне плоского графика ввести на оси абсцисс имя аргумента, на оси ординат – имя функции.
5. Щелкнуть мышью вне шаблона графика. График построен для заданного диапазона изменения аргумента.

Если диапазон значений аргумента не задан, по умолчанию график будет построен в диапазоне значений аргумента от -10 до 10.

Пример:

Построить график функции $f(x) = \sin(x)$

1. Ввод функции
 $f(x) := \sin(x)$
2. Выбрать кнопку построения плоского графика, затем ввести имена осей и щелкнуть вне шаблона графика.

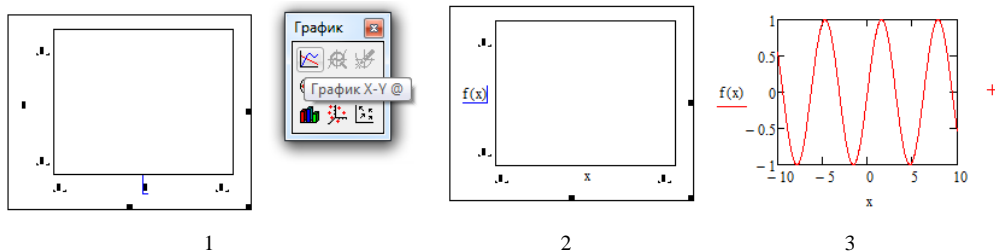


Рисунок 4.1 – Этапы построения графика, где 1 – выбор плоского графика, 2 – ввод названия осей, 3 – завершение построения графика

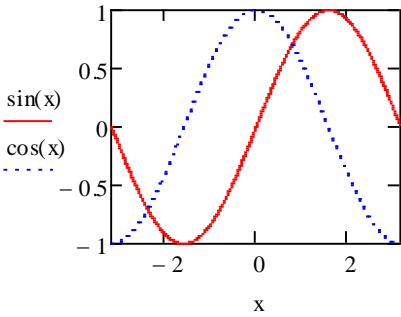


Рисунок 4.2 – Построение на одном графике функций синуса и косинуса

например, $f_1(x)$ и $f_2(y)$, то на оси ординат нужно ввести (через запятую) имена обеих функций, а на оси абсцисс (также через запятую) – имена обоих аргументов, x и y . Тогда первый график будет построен для первой функции по первому аргументу, а второй график – для второй функции по второму аргументу. Если функций введено несколько, а аргументов – 2, то график первой функции строится по первому аргументу, а графики остальных функций – по второму.

Функцию можно задать аналитически или массивами значений аргумента и функции. Для построения плоского графика массивы обязательно должны быть векторами одинаковой размерности. При задании функции массивами в местах ввода на осях графика надо вписать имена массивов с указанием индексов или без них.

ORIGIN:= 1
 ^^^^^^^^^^^

Data := $\begin{pmatrix} 2 \\ 5 \\ 8 \\ 14 \\ 20 \\ 16 \\ 12 \end{pmatrix}$

$i := 1..last(Data)$

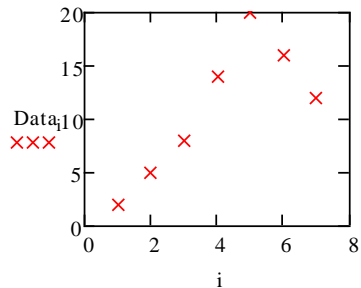


Рисунок 4.3 – Построение графика из массива данных

Чтобы в одном шаблоне разместить несколько графиков, необходимо, набрав на оси ординат имя первой функции, нажать клавишу «запятая» (уголок курсора при этом обязательно должен находиться в конце имени функции). В появившемся месте ввода (черном квадрате) вписать имя второй функции и т. д. Если две функции имеют разные аргументы,

1.1 Форматирование двумерных графиков

Чтобы отформатировать график необходимо сделать двойной щелчок мышью в поле графика – откроется окно форматирования графика (см. рисунок 4.4). Выбирая различные пункты меню (щелкнув на них мышью, а затем на кнопке *Применить* или *ОК*) можно заметить, как изменится при этом вид графика.

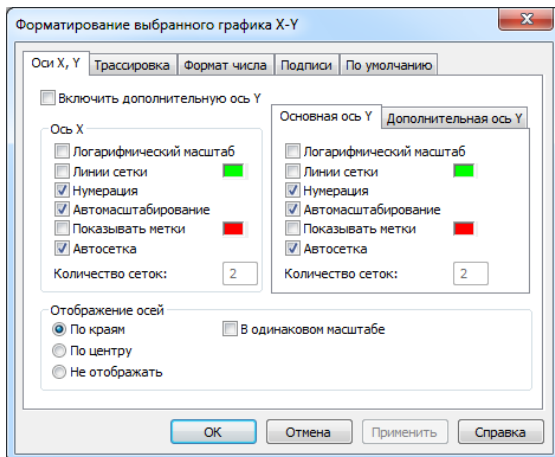


Рисунок 4.4 – Вкладка «Оси X,Y»

В открывшемся окне (см. рисунок 4.4) располагаются заголовки пяти вкладок:

1. *X-Y Axes (Оси X-Y)* – предназначена для форматирования осей координат: нанесения сетки линий, проставления численные значения; проведение ось абсцисс через ноль ординаты (*Crossed – По центру*) или по нижнему краю графика (*Boxed – По краям*), нанести метки на графике. Для выбора необходимых пунктов необходимо щелкните на них мышью (отметить их флажком).

- *Log Scale (Логарифмический масштаб)* – представить численные значения на осях в логарифмической шкале. По умолчанию численные значения наносятся в равномерной шкале;

- *Grid Lines (Линии сетки)* – нанести сетку линий;

- *Numbered (Нумерация)* – нанести численные значения для каждой линии сетки;

- *Auto Scale (Автомасштабирование)* – автоматически выбирает предельные численные значения на осях, большие максимальных вычисленных значений. Если этот пункт не отмечен, предельными будут максимальные вычисленные значения;

- *Show Markers (Показать метки)* – нанести метки на графике. На каждой оси появляются два места ввода, в которые можно ввести численные значения (можно не вводить ничего или ввести одно число или буквенные обозначения констант). На графике появятся горизонтальные или вертикальные пунктирные линии, соответствующие указанному значению на оси. В конце этой линии появляется само число.

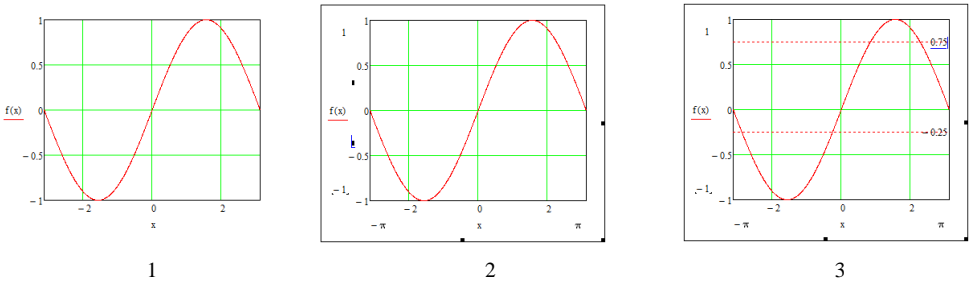


Рисунок 4.5 – Добавление меток на график, где 1 – изначальный график, 2 – добавление меток на оси Y, 3 – ввод данных и конечный результат

- *Autogrid (Автосетка)* – количество линий сетки выбирается автоматически. Если пункт не отмечен, надо задать количество линий Number of Grids (Количество сеток).

Имеется возможность нанести численные значения как слева, так и справа от графика. Если это необходимо, следует поставить флажок на пункте *Enable Secondary Y-axis (Включить дополнительную ось Y)*. С правой стороны графика появится *Placeholder (Место ввода)*, куда надо вписать имя функции, выводимой на график.

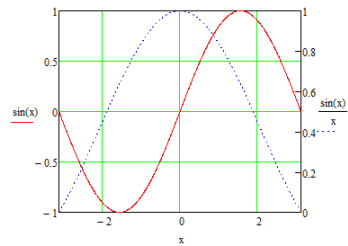


Рисунок 4.6 – Пример использования дополнительной оси

2. *Trace* (Трассировка) – служит для форматирования графиков функций.

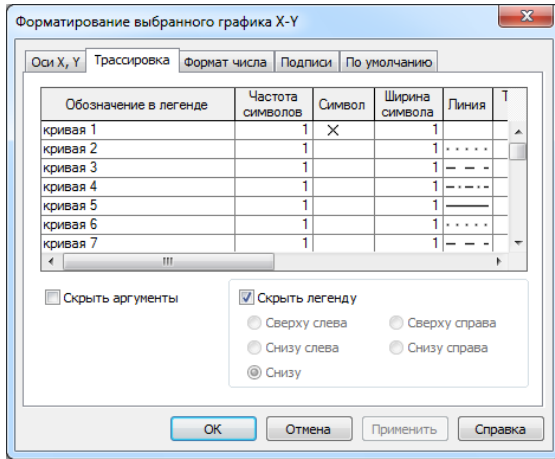


Рисунок 4.7 – Вкладка «Трассировка»

Для каждого графика в отдельности задаются:

- *Symbol Frequency* (Частота символов) – на графике можно отмечать не все расчетные точки, а через одну, через 2 и т. д.;
- *Symbol* (Символ) – обозначение расчетных точек на графике (выбрать круг, крест, прямоугольник, ромб);
- *Symbol Weight* (Ширина символа) – размер значка, которым обозначается расчетная точка;
- *Line* (Линия) – вид линии: *Solid* – сплошная, *Dot* – Пунктир, *Dash* – Длинный пунктир, *Dadot* – Штрих-пунктир;
- *Line Weight* (Толщина линии) – толщина линии;
- *Color* (Цвет) – цвет линии;
- *Type* (Тип) – тип графика: *Lines* – Линия, *Points* – Точки, *Bar* или *Solid-bar* – Столбы, *Step* – Ступенчатый график и др.;
- *Y-Axis* (Ось Y) – в этой колонке Mathcad сам проставляет Y или Y2 в зависимости от того, слева или справа будут выведены численные значения функции.

3. *Number Format (Формат чисел)* – формат чисел, выводимых на графике.

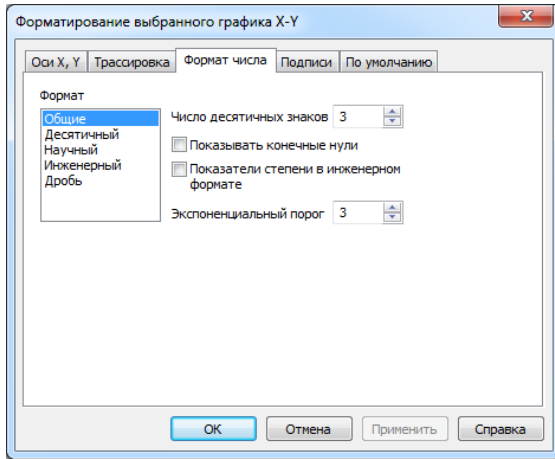


Рисунок 4.8 – Вкладка «Формат числа»

4. *Labels (Заголовки)* – вписать в область графика заголовков. В окне *Title (Заголовок)* нужно записать текст заголовка, выбрать его положение – сверху или внизу графика. Указать, если необходимо, названия осей (*Axis Label*).

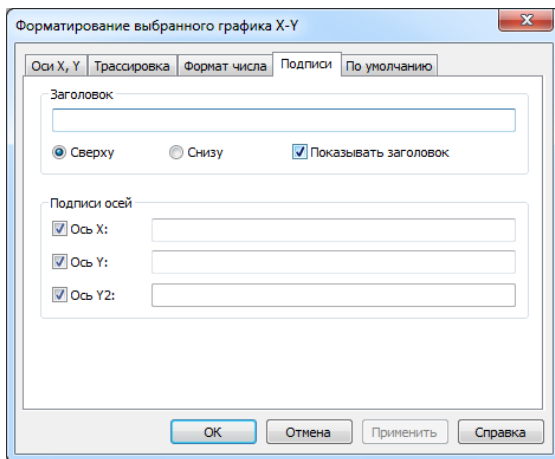


Рисунок 4.9 – Вкладка «Подписи»

5. *Defaults (По умолчанию)* – вернуться к виду графика, принятому по умолчанию (*Change to default*), либо использовать по умолчанию для всех графиков данного документа (*Use for default*) выполненные изменения.

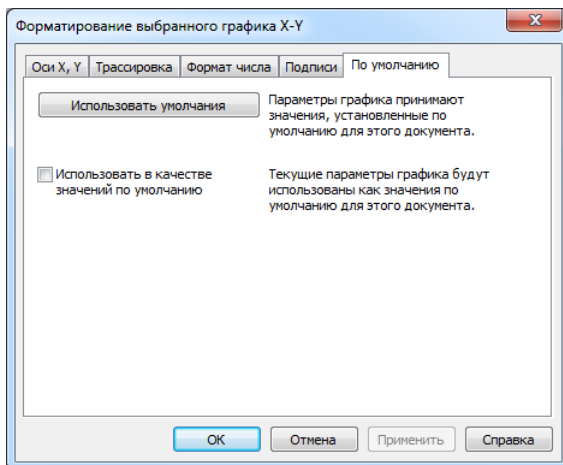


Рисунок 4.10 – Вкладка «По умолчанию»

1.2 Масштабирование графиков

Для изменения масштаба графика необходимо:

1. Поместить курсор в поле графика и нажать правую кнопку мыши. В открывшемся контекстном меню выбрать *Zoom (Масштаб)*.
2. Обвести пунктирной линией при нажатой левой кнопке мыши область графика, которую необходимо увеличить.
3. В открытом окне *X-Y Zoom (Масштаб по осям X-Y)* нажать кнопку *Zoom (Масштабировать)*. Выделенная область увеличится до размеров всей рамки графика.

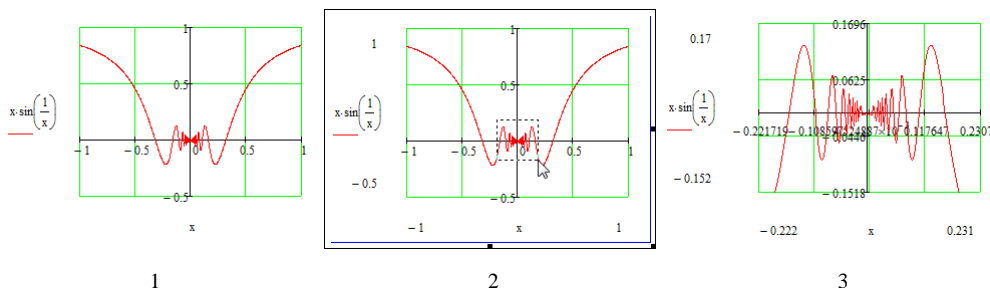


Рисунок 4.11 – Масштабирование графика, где 1 – изначальный график, 2 – выбор области масштабирования, 3 – конечный результат

Если необходимо вернуться к исходному виду графика, нужно нажать *Unzoom* (*Отменить масштабирование*) или *Full View* (*Вид полностью*). Если надо оставить увеличенное изображение, следует нажать кнопку ОК.

В контекстном меню, открываемом при нажатии правой кнопки мыши, есть пункт *Trace* (*Трассировка*). Выбор этого пункта вызывает появление окна трассировки X-Y Trace. Далее нужно щелкнуть левой кнопкой мыши на графике. На острие курсора появится перекрестье пунктирных линий, перемещающееся при движении курсора при нажатой левой кнопке мыши. Одновременно в окне X-Y Trace изменяются цифры, указывающие координаты перекрестья. Установив перекрестье на нужную точку графика, можно скопировать его координаты в буфер обмена и затем вставить в документ для использования в расчетах. Если поставлен флажок *Track Data Points* (*Отслеживать точки данных*), то перекрестье скачет от точки к точке. Если снять этот флажок, движение перекрестья будет плавным.

1.3 Изменение размеров и перемещение графиков

Чтобы изменить размеры графика, нужно:

1. Щелкнуть мышью в поле графика.
2. Подвести указатель мыши к одному из черных квадратов на краю графика.
3. При появлении двунаправленной стрелки нажать левую кнопку мыши и, удерживая ее, переместить край графика в нужное место.

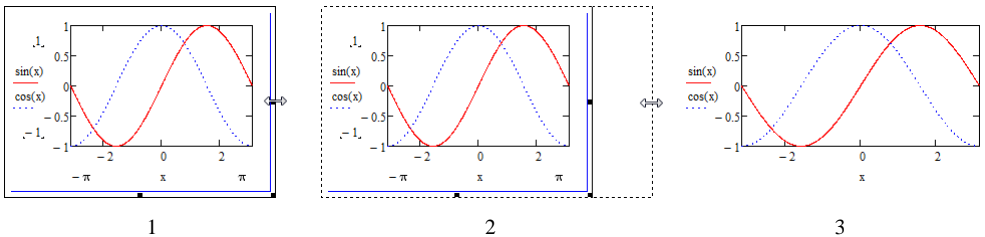


Рисунок 4.12 – Этапы изменения размера графика, где 1 – выделение графика, 2 – растягивание вдоль, 3 – конечный результат

Чтобы переместить график, нужно:

1. Щелкнуть мышью в поле графика;
2. Подвести указатель мыши к краю графика;
3. При появлении черной ладошки нажать левую кнопку мыши и, удерживая ее, переместить график в нужное место.

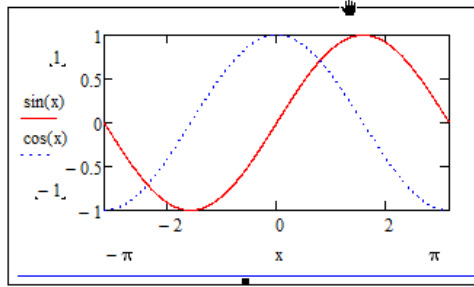


Рисунок 4.13 – Перемещение графика



Точно так же можно переместить любой объект документа: выражение, текст, график.

Если объекты (выражения, графики) перекрывают друг друга, в контекстном меню появляются пункты *Bring to Front* (*Выдвинуть на передний план*) и *Send to Back* (*Убрать на задний план*). Использование этих пунктов позволяет экономить место в документе, специально накладывая края объектов один на другой.

Если необходимо раздвинуть наложившиеся друг на друга объекты, следует выделить их, перечеркнув мышью при нажатой левой кнопке. Все выделенные объекты будут обведены пунктирной рамкой. В главном меню Mathcad следует выбрать *Format* → *Separate Region* (*Формат* → *Разделить области*). При этом будут раздвинуты только выделенные области.



Не следует нажимать кнопку *Separate Region*, не выделив предварительно объекты, так как Mathcad при этой команде раздвигает области сразу во всем документе. Для экономии места в документе объекты обычно располагают как можно ближе друг к другу. Mathcad раздвигает документы сверху вниз и при этом может нарушить созданную последовательность вычислений.

2 Построение полярных графиков

Чтобы построить график функции в полярных координатах необходимо:

1. В панели графиков *Graph Toolbar* щелкнуть на кнопке, изображающей полярный график.
2. В местах ввода ввести: внизу – угловой аргумент функции, слева – имя функции для построения обычного графика. Для построения параметрического графика нужно ввести в обоих местах ввода имена функций углового аргумента.

Форматирование полярных графиков аналогично форматированию плоских графиков. Полярные графики, как и плоские, можно строить в обычных и в логарифмических координатах.

Для построения графика в логарифмических координатах в окне форматирования графика надо отметить флажком пункт *Log Scale (Логарифмический масштаб)*.

Пример:

Построить в полярных координатах с шагом в 6 градусов график функции $f(\alpha) = \cos(2 \cdot \alpha)$

1. Ввод диапазона

$$\alpha := 0, \frac{\pi}{30} \dots 2 \cdot \pi$$

2. Создание графика функции

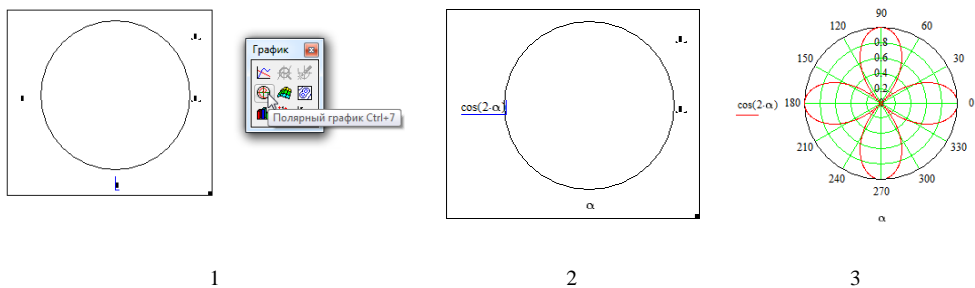


Рисунок 4.17 – Этапы построения графика в полярных координатах, где 1 – выбор графика поверхности, 2 – ввод функции и углового аргумента, 3 – завершение построения графика

3 Построение трехмерных графиков

Для построения трехмерного графика необходимо:

1. Набрать имя функции двух переменных, знак присвоения значения ($:=$) и выражение функции.
2. Установить курсор в то место, где необходимо построить график.
3. В математической панели щелкнуть мышью на кнопке *Graph Toolbar* (Панель графиков), изображающей график, затем на *Surface Plot* (График поверхности). На месте курсора появится шаблон трехмерного графика.
4. В единственном поле ввода шаблона графика ввести имя функции.
5. Щелкнуть мышью вне области шаблона. График построен.

Пример:

Построить график функции $f(x, y) = x^2 \cdot y^2$

1. Ввод функции

$$f(x, y) := x^2 \cdot y^2$$

2. Выбрать кнопку построения графика поверхности, затем ввести имя функции и щелкнуть вне шаблона графика.

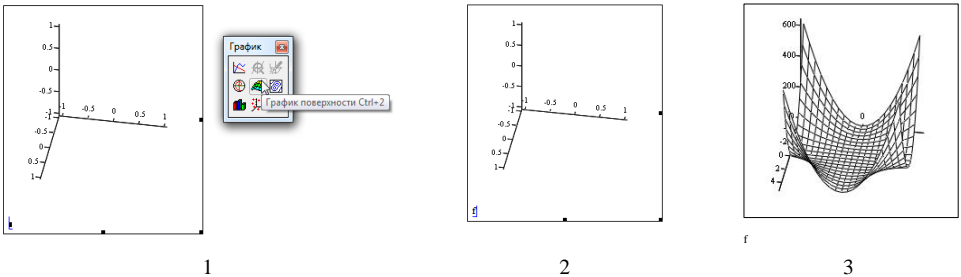
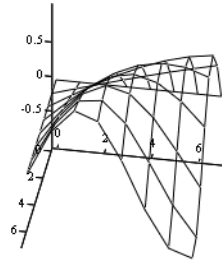


Рисунок 4.14 – Этапы построения графика поверхности, где 1 – выбор графика поверхности, 2 – ввод имени функции, 3 – завершение построения графика

Кроме описанного способа ускоренного построения графика поверхности по функции двух переменных существует и часто применяется другой способ создания графика поверхности – с использованием массива численных значений функции. Для этого необходимо:

1. С помощью дискретных переменных ввести значения обоих аргументов заданной функции.
2. Ввести массив, элементами которого являются значения функции, вычисленные при заданных значениях аргументов.
3. Повторить п. 3-5 последовательности построения трехмерного графика

$$\begin{aligned}
 i &:= 0..7 & j &:= 0..7 \\
 X_i &:= 0.5i & Y_j &:= 0.2j \\
 Z_{i,j} &:= \sin(X_i \cdot Y_j)
 \end{aligned}$$



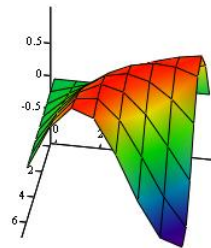
z

Рисунок 4.15 – Построение графика поверхности при помощи использования массива численных значений

3.1 Форматирование трехмерных графиков

Чтобы попасть в окно форматирования графика необходимо сделать двойной щелчок мышью в поле графика. В данном окне можно настроить:

- Выбрав пункт *Appearance* → *Fill surface* → *Colormap* (*Оформление* → *Поверхность с заливкой* → *Карта цветов*) и щелкнув мышью на кнопке *Применить* можно сделать график цветным.



z

Рисунок 4.16 – Пример цветного графика

- Для придания эффекта освещения нужно выбрать в окне форматирования пункт *Lighting* → *Enable Lighting* → *On* (*Подсветка* → *Необходима подсветка* → *Включить подсветку*). Далее нужно выбрать схему освещения (*Lighting schema*). В выпадающем меню находится 6 различных схем.

4 Поверхности тел вращения

Аналитически тела вращения, как правило, описываются в сферических или цилиндрических координатах.

Пример:

Построить шар с вырезом при помощи использования аналитического преобразования координат

1. Задание числа расчетных точек и ввод индексов

$$n := 40 \quad i := 1..n \quad j := 1..n$$

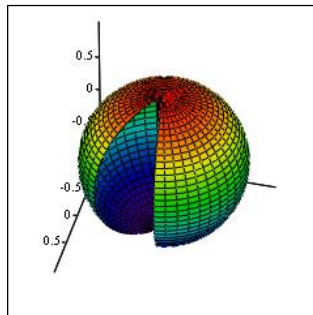
2. Ввод функций изменения углов

$$\alpha_i := i \cdot \frac{\pi}{n} \quad \beta_j := j \cdot 2 \cdot \frac{\pi}{n + 5}$$

3. Ввод функций расчета координат шара

$$X_{i,j} := \sin(\alpha_i) \cdot \cos(\beta_j) \quad Y_{i,j} := \sin(\alpha_i) \cdot \sin(\beta_j) \quad Z_{i,j} := \cos(\alpha_i)$$

4. Полученный график



(X, Y, Z)

Рисунок 4.17 – Шар построенный при помощи аналитического преобразования координат

Пример:

Построить фигуру, полученную вращением относительно оси графика функции $f(x) = x \cdot \sin^2(x)$

1. Ввод функции

$$f(x) := x \sin(x)^2$$

2. Ввод диапазона построения графика функции
 $x := -\pi, -\pi + 0.01..2\pi$
3. Построение графика образующей

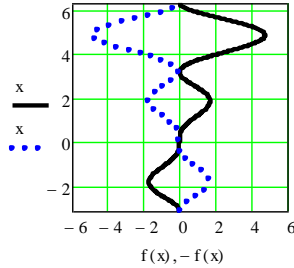


Рисунок 4.18 – Прямой и обратный график функции $f(x) = x \cdot \sin^2(x)$

4. Преобразование уравнения функции в цилиндрические координаты

$$\underline{\underline{S}}(\underline{u}, v) := \begin{pmatrix} u \\ f(u) \cdot \cos(v) \\ f(u) \cdot \sin(v) \end{pmatrix}$$

5. Полученный результат

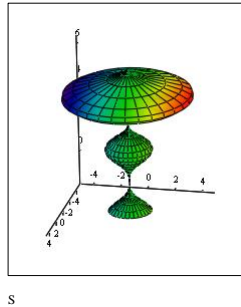


Рисунок 4.19 – Фигура, полученная вращением относительно оси графика функции $f(x) = x \cdot \sin^2(x)$

5 Построение сложных фигур

Построение пространственной кривой с применением функции **CreateSpace**. Встроенная функция **CreateSpace** принимает трехэлементную векторную

функцию одной переменной и возвращает декартовы координаты пространственной кривой.

Обращение к функции:

CreateSpace(F, t0, t1, tgrid, fmap),

где **F** – трехэлементная векторная функция одной переменной; **t0** и **t1** – пределы изменения переменной, по умолчанию от -5 до 5; **tgrid** – число точек кривой, по умолчанию 20; **fmap** – функция преобразования координат. Параметр **fmap** необязателен. При его отсутствии входная функция по умолчанию считается заданной в декартовых координатах.

```
H(t) := (sqrt(t) * cos(t),
sqrt(t) * sin(t),
t)
t0 := 0 t1 := 10
C := CreateSpace(H, t0, t1, 200)
```

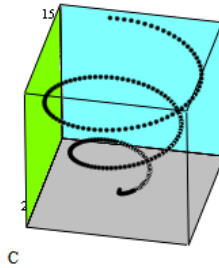


Рисунок 4.20 – Фигура, построенная при помощи функции **CreateSpace**

6 Пересекающиеся фигуры

n := 40 i := 1..n j := 1..n

$X_{i,j} := \cos\left(\frac{2 \cdot \pi \cdot i}{n}\right)$ $Y_{i,j} := \sin\left(\frac{2 \cdot \pi \cdot i}{n}\right)$

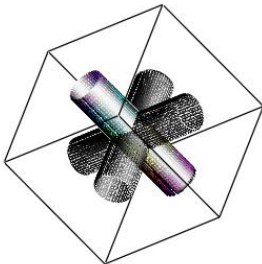
$Z_{i,j} := \frac{j - 20}{5}$

i := 0..N $\phi_i := i \cdot 2 \cdot \frac{\pi}{N}$ j := 0..N $\alpha_j := j \cdot 2 \cdot \frac{\pi}{N}$

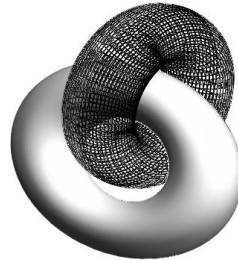
$x_{1,j} := (5 + 2 \cdot \cos(\phi_i)) \cdot \cos(\alpha_j)$

$y_{1,j} := (5 + 2 \cdot \cos(\phi_i)) \cdot \sin(\alpha_j)$

$z_{1,j} := 2 \cdot \sin(\phi_i)$



(X, Y, Z), (Z, Y, X), (Y, Z, X)



(x, y + 6, z), (x, y + 1, z)

Рисунок 4.21 – Примеры пересекающихся поверхностей

7 Построение многогранников

В Mathcad имеется возможность быстрого построения поверхностей многогранников. Всего можно построить 80 многогранников. Описание многогранников, которые Mathcad может вывести на экран, можно найти в Центре документации: *Resource Center* → *QuickSheets and Reference Tables* → *Reference Tables* → *Geometry Formulas* → *Polyhedra* (Центр ресурсов → Таблица справок → Геометрические формулы → Многогранники).

Например, для построения полиэдра необходимо вывести на экран шаблон 3D-графика и в поле ввода написать **Polyhedron** ("**#«номер от 1 до 80»**"). Примеры построения многогранников приведены на рисунке ниже.

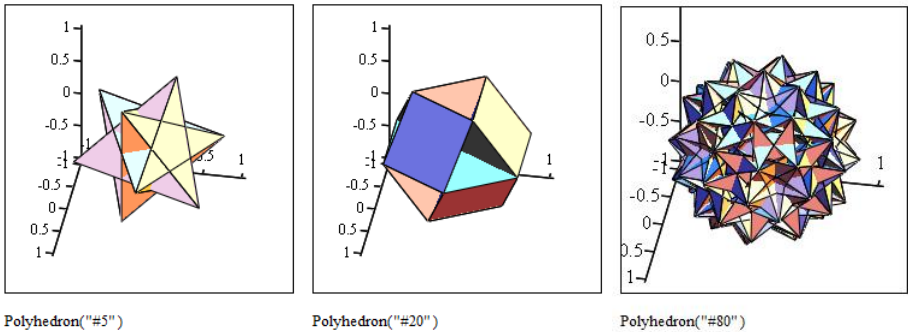


Рисунок 4.22 – Примеры построения многогранников

8 График векторного поля

Векторное представление функции создается построением коротких стрелочек – векторов. Стрелочка направлена в сторону возрастания высоты поверхности (увеличения значений функции). Плотность стрелок зависит от скорости изменения функции, то есть от ее градиента. Длина стрелочки пропорциональна модулю вектора. Такого вида графики удобно приводить при расчете различных полей: электромагнитных, тепловых, гравитационных и т. д.

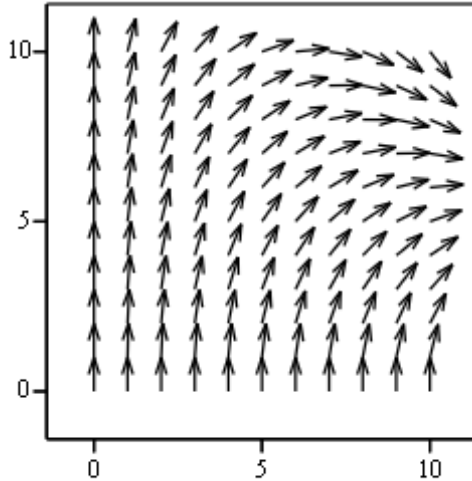
При построении векторного поля требуется задать двухмерный массив значений функции в заданных точках. На рисунке 4.22 показано векторное поле – поток функции G , у которой проекции на осях X и Y – это H_1 и H_2 соответственно. Иногда удобно представить функцию как матрицу комплексных чисел, где

одна из проекций вектора – его вещественная часть, другая проекция – мнимая часть.

$$F1(x,y) := \sin(x \cdot y) \quad F2(x,y) := \cos(x \cdot y) \quad n := 10 \quad m := 10$$

$$i := 0..n \quad j := 0..m \quad x_i := \frac{\pi}{2 \cdot n} \cdot i \quad y_j := \frac{\pi}{2 \cdot m} \cdot j$$

$$H1_{i,j} := F1(x_i, y_j) \quad H2_{i,j} := F2(x_i, y_j) \quad G_{i,j} := H1_{i,j} + i \cdot H2_{i,j}$$



(H1, H2)

Рисунок 4.23 – Пример построения графика векторного поля

Обработка экспериментальных данных

1 Функции для обработки экспериментальных данных

Многим из тех, кто сталкивается с научными и инженерными расчётами часто приходится оперировать наборами значений, полученных опытным путём или методом случайной выборки. Как правило, на основании этих наборов требуется построить функцию, на которую могли бы с высокой точностью попадать другие получаемые значения. Если кривая, описываемая этой функцией, должна проходить через все экспериментальные точки, операция получения промежуточных точек и расчетной функции называется интерполяцией.

Если кривая, описываемая этой функцией, не должна проходить через все экспериментальные точки и является аппроксимацией (усреднением) исходных данных, операция получения промежуточных точек и расчетной функции называется регрессией. Интерполяция – частный случай более общей задачи аппроксимации (приближенного представления), возникающей при замене кривой, описываемой функцией сложной природы, другой кривой, в некотором смысле близкой заданной, имеющей более простые уравнения. Интерполяция является способом нахождения промежуточных значений величины по имеющемуся дискретному набору известных значений.

Если необходимо уменьшить разброс данных или исключить некоторую систематическую погрешность, например, в виде наложенных колебаний, используют сглаживание данных или фильтрацию спектра колебаний данных.

Задача сглаживания кривой возникает, когда данные, используемые для ее восстановления, определены в результате измерений или эмпирически с некоторой погрешностью либо представляет кривую, описываемую функцией, недостаточно гладкой (например, недифференцируемой или дифференцируемой всего несколько раз).

1.1 Интерполяция

Интерполяция использует значения некоторой функции, заданные в ряде точек, чтобы предсказать значения функции между ними.

1.1.1 Линейная интерполяция

При линейной интерполяции Mathcad соединяет существующие точки данных прямыми линиями. Это выполняется функцией **linterp**.

Функция **linterp(X, Y, x)** использует векторы данных X и Y , чтобы вернуть линейно интерполируемое значение y , соответствующее третьему аргументу x . Аргументы X и Y должны быть векторами одинаковой длины. Вектор X должен содержать вещественные значения, расположенные в порядке возрастания. Эта функция соединяет точки данных отрезками прямых, создавая таким образом ломаную. Интерполируемое значение для конкретного x есть ордината y соответствующей точки ломаной.

Пример:

Провести линейную интерполяцию экспериментальных данных:

| | | | | | |
|-------|-----|-----|-----|-----|------|
| F, Н | 4,7 | 5,0 | 5,2 | 5,4 | 5,9 |
| S, мм | 3,0 | 3,0 | 4,0 | 7,0 | 10,0 |

и определить значение S при $F=5,6$ Н. Построить график и указать на нем найденную точку.

1. Ввод данных в программу

$$\underline{\underline{F}} := (4.7 \ 5 \ 5.2 \ 5.4 \ 5.9) \quad \underline{\underline{S}} := (3 \ 3 \ 4 \ 7 \ 10)$$

2. Преобразование данных в вектора

$$F := F^T \quad S := S^T$$

3. Задание точки F для поиска расчетного значения

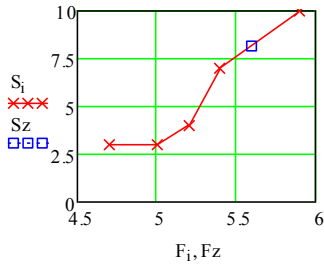
$$Fz := 5.6$$

4. Определение значения S в данной точке.

$$Sz := \text{linterp}(F, S, Fz) \quad Sz = 8.2$$

5. Построение графика и указание расчетной точки.

$$i := \text{ORIGIN}.\text{last}(F)$$



1.1.2 Сплайн-интерполяция

Гораздо лучшие результаты дает сплайн-интерполяция. При ней исходная функция заменяется отрезками кубических полиномов, проходящих через три смежные узловые точки. Коэффициенты полиномов рассчитываются так, чтобы непрерывными были первая и вторая производные. Линия, которую описывает сплайн-функция, напоминает по форме гибкую линейку, закрепленную в узловых точках (отсюда и название интерполяции: spline – гибкая линейка).

Для осуществления сплайновой интерполяции система MathCAD предлагает четыре встроенные функции. Три из них служат для получения векторов вторых производных сплайн-функций при различном виде интерполяции:

- **cspline(X, Y)** – возвращает вектор вторых производных при приближении в опорных точках к кубическому полиному;
- **pspline(X, Y)** – возвращает вектор вторых производных при приближении к опорным точкам параболической кривой;
- **lspline(X, Y)** – возвращает вектор вторых производных при приближении к опорным точкам прямой;
- **interp(S, X, Y, x)** – возвращает значение $y(x)$ для заданных векторов S, X, Y и заданного значения x.

Таким образом, сплайн-интерполяция проводится в два этапа. На первом с помощью функций **cspline**, **pspline** или **lspline** отыскивается вектор вторых производных функции $y(x)$, заданной векторами X и Y ее значений (абсцисс и ординат). Затем, на втором этапе для каждой искомой точки вычисляется значение $y(x)$ с помощью функции **interp**.

Пример:

Провести сплайн-интерполяцию с различными видами экстраполяции экспериментальных данных:

| | | | | | |
|-------|-----|-----|-----|-----|------|
| X, мм | 4,7 | 5,0 | 5,2 | 5,4 | 5,9 |
| Y, мм | 3,0 | 3,0 | 4,0 | 7,0 | 10,0 |

и определить значение Y при X=5,75. Построить графики и указать на них найденные точки.

1. Ввод данных в программу

$$X := (4.7 \ 5 \ 5.2 \ 5.4 \ 5.9) \quad Y := (3 \ 3 \ 4 \ 7 \ 10)$$

2. Преобразование данных в вектора

$$F := F^T \quad S := S^T$$

3. Задание точки для поиска расчетного значения

$$xz := 5.75$$

4. Определение значения функции в данной точке.

$$yc := \text{interp}(D2C, X, Y, xz) \quad yc = 11.263$$

$$yp := \text{interp}(D2P, X, Y, xz) \quad yp = 10.072$$

$$yl := \text{interp}(D2L, X, Y, xz) \quad yl = 9.676$$

5. Построение графиков и указание расчетной точки.

$$h := 0.05$$

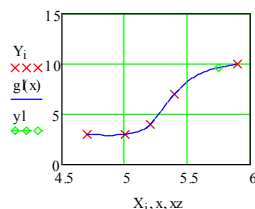
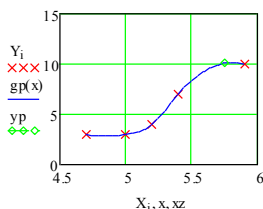
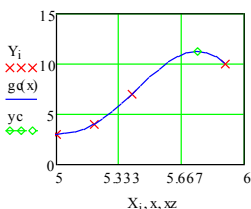
$$x := \text{min}(X), \text{min}(X) + h .. \text{max}(X)$$

$$gc(x) := \text{interp}(D2C, X, Y, x)$$

$$gp(x) := \text{interp}(D2P, X, Y, x)$$

$$gl(x) := \text{interp}(D2L, X, Y, x)$$

$$i := \text{ORIGIN} . \text{last}(X)$$



1.2 Регрессия

Другой широко распространенной задачей обработки данных является представление их совокупности некоторой функцией $y(x)$. Задача регрессии заключается в получении параметров этой функции такими, чтобы функция приближала облако исходных точек (заданных векторами VX и VY) с наименьшей среднеквадратичной погрешностью. Регрессия сводится к подбору коэффициентов в той или иной аналитической зависимости.

В Mathcad регрессия с использованием одного полинома реализуется комбинацией встроенных функций регрессии и интерполяции

interp(s, x, y, t); regress(x, y, n)

где x – вектор значений аргумента, элементы которого расположены в порядке возрастания; y – вектор значений функции того же размера; s – вектор коэффициентов для построения аппроксимирующего полинома, создаваемый функцией `regress`; t – значение аргумента, при котором вычисляется интерполирующая функция; n – степень аппроксимирующего полинома.

Степень аппроксимирующего полинома может быть любой. Практика показывает, что полинома 5-й степени достаточно для аппроксимации почти любой кривой.

Обращение к указанным функциям:

$s:=\text{regress}(X,Y,n)$, $YY(t):=\text{interp}(s,X,Y,t)$ или $YY(t):=\text{interp}(\text{regress}(X,Y,n),X,Y,t)$

Регрессия с использованием нескольких отрезков полинома реализуется комбинацией встроенных функций регрессии и интерполяции **interp(s, x, y, t)** и **loess(x, y, span)**, где **$s:=\text{loess}(X, Y, \text{span})$** – вектор коэффициентов для построения аппроксимирующего полинома 2-й степени, требуемый функцией **interp**; **span>0** – параметр, определяющий размер отрезков полиномов.

Параметр `span` задает степень сглаженности данных. На практике $0,2 < \text{span} < 2$. При `span=2` результат аппроксимации тот же, что и при аппроксимации одной параболой. При `span=0,2` аппроксимирующая кривая почти точно описывает любой набор данных.

Пример:

Провести регрессию полиномом 3-ей степени и отрезками полинома 2-ой степени экспериментальных данных с построением графика:

| | | | | | | | | |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| № | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| Y | 38 | 15 | 5,5 | 23 | 51 | 79 | 105 | 126 |
| ϕ | 0,000 | 0,349 | 0,698 | 1,047 | 1,396 | 1,745 | 2,094 | 2,443 |

Продолжение таблицы

| | | | | | | | | |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| № | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| Y | 144 | 156 | 164 | 166 | 152 | 137 | 117 | 93 |
| ϕ | 2.793 | 3,142 | 3,491 | 3,840 | 4,189 | 4,538 | 4,887 | 5,236 |

1. Ввод данных в программу

$Y := (38 \ 15 \ 5.5 \ 23 \ 51 \ 79 \ 105 \ 126 \ 144 \ 156 \ 164 \ 166 \ 152 \ 137 \ 117 \ 93)$

$\phi := (0 \ 0.349 \ 0.698 \ 1.047 \ 1.396 \ 1.745 \dots 4.887 \ 5.236)$

2. Преобразование данных в вектора

$Y := Y^T \quad \phi := \phi^T$

3. Регрессия полиномом 3-ей степени

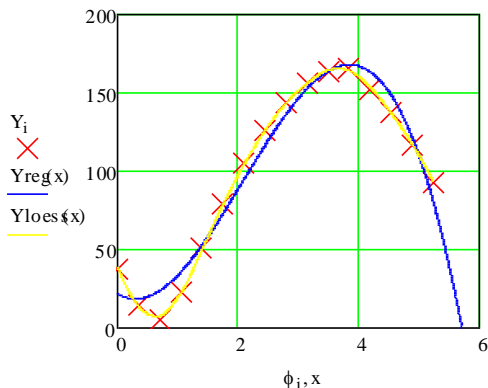
$Y_{reg}(x) := \text{interp}(\text{regress}(\phi, Y, 3), \phi, Y, x)$

4. Регрессия отрезками полинома второй степени с длиной отрезка 0,4

$\text{span} := 0.4 \quad Y_{loess}(x) := \text{interp}(\text{loess}(\phi, Y, \text{span}), \phi, Y, x)$

5. Построение графика

$i := 0..15$



1.2.1 Обобщенная регрессия

Линейная или полиномиальная функции не во всех случаях подходят для описания зависимости данных. Бывает, что нужно искать эту зависимость в виде линейных комбинаций произвольных функций, ни одна из которых не является полиномом. Например, в рядах Фурье следует аппроксимировать данные, используя линейную комбинацию комплексных экспонент.

Функция **linfit** разработана, чтобы решить эти виды проблем. Если предполагается, что данные могли бы быть смоделированы в виде линейной комбинации произвольных функций вида:

$$f(x) = a_0 \cdot f_0(x) + a_1 \cdot f_1(x) + \dots a_n \cdot f_n(x)$$

следует использовать **linfit**, чтобы вычислить a_i .

Функция **linfit (X, Y, F)** возвращает вектор, содержащий коэффициенты, используемые, чтобы создать линейную комбинацию функций из F, дающую наилучшую аппроксимацию данных из векторов X и Y. F – функция, которая возвращает вектор, состоящий из функций, которые нужно объединить в виде линейной комбинации.

Пример:

Провести регрессию экспериментальных данных с построением графика зависимости вида $f(x) = a_0 \cdot x^2 + a_1 \cdot x + a_2 \cdot \frac{1}{1+x}$:

| | | | | | | |
|-------|------|------|------|------|------|------|
| X, мм | 0 | 0,2 | 0,4 | 0,6 | 0,8 | 1,0 |
| Y, мм | 0,43 | 0,22 | 0,80 | 0,10 | 1,00 | 2,00 |

1. Ввод данных в программу

X := (0 0.2 0.4 0.6 0.8 1)

Y := (0.43 0.22 0.80 0.1 1 2)

$$F(x) := \begin{pmatrix} x^2 \\ x \\ \frac{1}{1+x} \end{pmatrix}$$

2. Преобразование данных в вектора

$$X := X^T \quad Y := Y^T$$

3. Определение коэффициентов уравнения

$$S := \text{linfit}(X, Y, F)$$

$$S = \begin{pmatrix} 3.087 \\ -1.475 \\ 0.515 \end{pmatrix}$$

$$a0 := S_0 \quad a0 = 3.087$$

$$a1 := S_1 \quad a1 = -1.475$$

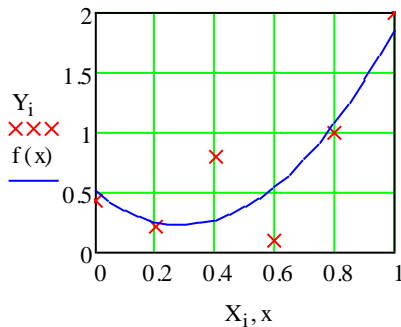
$$a2 := S_2 \quad a2 = 0.515$$

4. Построение графика

$$x := \text{min}(X), \text{min}(X) + 0.05.. \text{max}(X)$$

$$f(x) := F(x) \cdot S$$

$$i := 0.. \text{length}(X)$$



Имеются случаи, когда гибкость **linfit** недостаточна – данные должны быть смоделированы не линейной комбинацией данных, а некоторой функцией, чьи параметры должны быть выбраны. Например, если данные могут быть смоделированы в виде суммы

$$f(x) = a_0 \cdot \sin(2x) + a_1 \cdot \tanh(3x)$$

То нужно решить уравнение относительно неизвестных коэффициентов a_0 и a_1 , значит эта проблема решается с помощью **linfit**.

Если данные должны быть смоделированы в виде суммы

$$f(x) = 2 \cdot \sin(a_1 x) + 3 \cdot \tanh(a_2 x)$$

и требуется найти неизвестные параметры a_1 и a_2 , то это задача для функции **genfit**.

Функция **genfit(X, Y, G, F)** возвращает вектор, содержащий n параметров u_0, u_1, \dots, u_{n-1} , которые обеспечивают наилучшее приближение данных из X и Y функцией f , зависящей от x и параметров u_0, u_1, \dots, u_{n-1} . F – функция, которая возвращает $n+1$ -мерный вектор, содержащий f и ее частные производные относительно параметров. G есть n -мерный вектор начальных значений для n параметров.

Пример:

Провести регрессию экспериментальных данных с построением графика зависимости вида $f(x) = e^{u_0 + u_1 \cdot x + u_2 \cdot x^2}$:

| | | | | | | |
|-------|-----|------|-----|-----|-----|-----|
| X, мм | 0,3 | 0,4 | 1,0 | 1,4 | 2,0 | 4,0 |
| Y, мм | 9,4 | 11,2 | 5,0 | 3,0 | 6,0 | 0 |

1. Ввод данных в программу

$X := (0.3 \ 0.4 \ 1 \ 1.4 \ 2 \ 4)$

$Y := (9.4 \ 11.2 \ 5 \ 3 \ 6 \ 0)$

$$F(z, u) := \begin{pmatrix} e^{u_0 + u_1 \cdot z + u_2 \cdot z^2} \\ e^{u_0 + u_1 \cdot z + u_2 \cdot z^2} \\ z \cdot e^{u_0 + u_1 \cdot z + u_2 \cdot z^2} \\ \frac{2}{z} \cdot e^{u_0 + u_1 \cdot z + u_2 \cdot z^2} \end{pmatrix}$$

2. Преобразование данных в вектора

$$X := X^T \quad Y := Y^T$$

3. Задание вектора начальных приближений

$$G := \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$$

4. Определение коэффициентов уравнения

$$P := \text{genfit}(X, Y, G, F)$$

$$P = \begin{pmatrix} 2.565 \\ -0.788 \\ 0.036 \end{pmatrix}$$

$$u0 := P_0 \quad u0 = 2.565$$

$$u1 := P_1 \quad u1 = -0.788$$

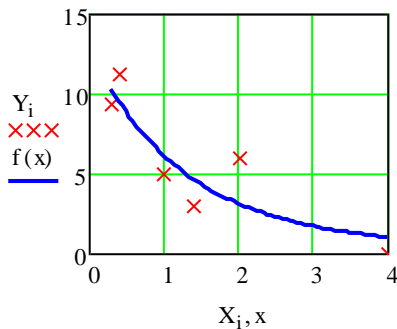
$$u2 := P_2 \quad u2 = 0.036$$

5. Построение графика

$$x := \text{min}(X), \text{min}(X) + 0.05.. \text{max}(X)$$

$$f(x) := F(x, P)_0$$

$$i := 0.. \text{length}(X)$$



Различие между функциями **linfit** и **genfit** есть различие между решением системы линейных уравнений и решением системы нелинейных уравнений. Первая задача легко решается методами линейной алгебры. Вторая задача гораздо более трудная и решается итеративными методами. Это объясняет, почему **genfit** нуждается в векторе начальных значений в качестве аргумента, а **linfit** в этом не нуждается.

1.2.2 Получение аналитического выражения аппроксимирующей функции

В Mathcad имеется восемь встроенных функций для получения аналитического выражения аппроксимирующей функции.

Необходимо использовать соответствующий вид регрессии, если хорошо известно, какой зависимостью описывается массив данных. Результат часто оказывается неудовлетворительным, когда вид регрессии плохо соответствует набору данных, и сделан плохой выбор начальных приближений.

Из восьми встроенных функций пять требуют предварительного задания вектора начальных приближений:

- **expfit**(X, Y, g) – регрессия экспонентой $f(t) = a \cdot e^{b \cdot t} + c$;
- **sinfitt**(X, Y, g) – регрессия синусоидой $f(t) = a \cdot \sin(t + b) + c$;
- **pwrfit**(X, Y, g) – регрессия степенной зависимостью $f(t) = a \cdot t^b + c$;
- **lgsffit**(X, Y, g) – регрессия логистической функцией $a(e) = \frac{a}{1 + b \cdot e^{-c \cdot t}}$;
- **logffit**(X, Y, g) – регрессия логарифмической функцией

$$f(t) = a \cdot \ln(t + b) + c$$

В этих функциях x – вектор значений аргумента, элементы которого расположены в порядке возрастания; y – вектор значений функции того же размера; g – вектор начальных приближений коэффициентов a, b и c; t – значение аргумента, при котором вычисляется интерполирующая функция.

Функции, не требующие начальных приближений:

- **line**(X, Y) – регрессия прямой линией, использующая минимизацию суммы квадратов ошибок $f(t) = a + bt$;
- **medfit**(X, Y) – регрессия прямой линией, использующая медиан-медианную линейную регрессию $f(t) = a + bt$. Функции **line** и **medfit** дают близкие результаты, слегка различающиеся наклоном прямых линий;
- **lnfit**(X, Y) – регрессия логарифмической функцией $f(t) = a \cdot \ln(t) + b$.

Для оценки связи между массивом данных и значениями аппроксимирующей функции подсчитан коэффициент корреляции **corr**. Чем выше данный коэффициент (чем больше он приближается к 1), тем удачнее проведена аппроксимация.

Пример:

Провести регрессию экспонентой экспериментальных данных с определение коэффициентов уравнения графика результирующей функции с определением коэффициента корреляции

| | | | | | |
|-------|-----|-----|-----|-----|-----|
| X, мм | 4,7 | 5,0 | 5,2 | 5,4 | 5,9 |
| Y, мм | 3 | 3 | 4 | 7 | 10 |

1. Ввод данных в программу и вектора начальных приближений

$$X := (4.7 \ 5 \ 5.2 \ 5.4 \ 5.9) \quad Y := (3 \ 3 \ 4 \ 7 \ 10)$$

$$g_0 := (1 \ 1 \ 1)$$

2. Преобразование данных в вектора

$$X := X^T \quad Y := Y^T \quad g := g^T$$

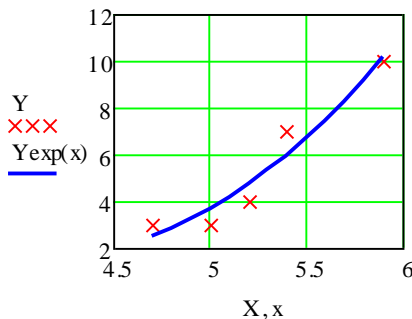
3. Определение коэффициентов уравнения

$$C := \text{expfit}(X, Y, g) = \begin{pmatrix} 0.055 \\ 0.908 \\ -1.407 \end{pmatrix}$$

4. Построение графика и определение коэффициента корреляции

$$Y_{\text{exp}(x)} := C_0 \cdot e^{C_1 \cdot x} + C_2$$

$$x := \min(X), \min(X) + 0.1.. \max(X)$$



$$\text{corr}(Y_{\text{exp}(X)}, Y) = 0.967$$

1.3 Функции сглаживания

Сглаживание предполагает использование набора значений y (и возможно x) и возвращение нового набора значений y , который является более гладким, чем исходный набор. В отличие от регрессии и интерполяции, сглаживание приводит к новому набору значений y , а не к функции, которая может оценивать значения между заданными точками данных.

Функция **ksmooth** использует гауссово ядро, чтобы вычислить локально взвешенные средние значения исходного вектора Y . Это сглаживание наиболее полезно, когда данные взяты в точках, отделяемых друг от друга интервалами приблизительно равной ширины. Если длина интервалов существенно изменяется, следует использовать адаптивное сглаживание подобное **supsmooth**.

ksmooth(X, Y, b) – возвращает n -мерный вектор, созданный сглаживанием при помощи гауссова ядра данных из Y . Y и X – n -мерные векторы вещественных чисел. Параметр b управляет окном сглаживания и должен быть установлен в несколько раз больше величины интервала между точками x .

Функция **supsmooth** использует симметричную линейную процедуру сглаживания методом наименьших квадратов по правилу k -ближайших соседей, чтобы выполнить локальную линейную аппроксимацию исходных данных. В отличие от **ksmooth**, который использует фиксированную ширину полосы сглаживания для всех исходных данных, **supsmooth** адаптивно выбирает различную ширину полосы сглаживания для различных частей данных.

supsmooth(X, Y) – возвращает n -мерный вектор, созданный локальным использованием симметричной линейной процедуры сглаживания методом наименьших квадратов по правилу k -ближайших соседей, в которой k выбирается адаптивно. Y и X – n -мерные векторы вещественных чисел. Элементы X должны быть расположены в порядке возрастания.

Функция **medsmooth** – наиболее устойчивая из трех, так как в меньшей степени подвержена влиянию зашумленных данных. Эта функция использует сглаживание с помощью скользящей медианы, вычисляет остатки, сглаживает остатки тем же самым образом и суммирует эти два сглаженных вектора вместе.

medsmooth(Y, n) – возвращает m-мерный вектор, созданный сглаживанием Y с помощью скользящей медианы. Y есть m-мерный вектор вещественных чисел. n – ширина окна, по которому происходит сглаживание. n должно быть нечетным числом, меньшим, чем число элементов в Y.

Пример:

Провести сглаживание данных, полученные в 120 расчетных точках из функции синусоиды, которая претерпевает случайное изменение с амплитудой 0,2. Шаг измеряемой функции – $\pi/10$. Построить график.

1. Ввод числа расчетных точек и шага измеряемой функции

$$i := 0..120 \quad x_i := \frac{\pi}{10} \cdot i$$

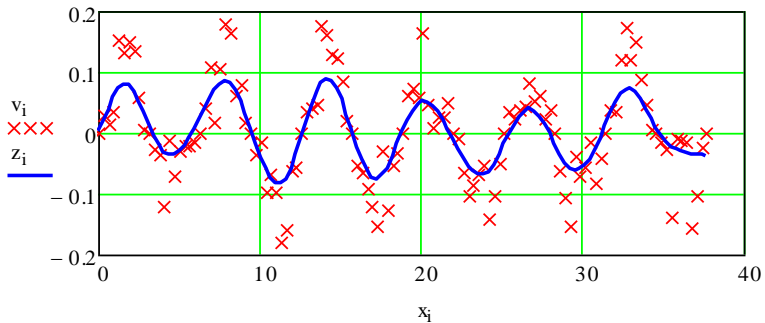
2. Задание измеряемой функции

$$v_i := \sin(x_i) \cdot \text{rnd}(0.2)$$

3. Сглаживание полученных данных

$$z := \text{supsmooth}(x, v)$$

4. Построение графика



Программирование и анимация

1 Программирование

Раздел «Программирование» занимает особое место. Большие возможности Mathcad позволяют решать подавляющее число задач без использования программирования, как правило, несколькими способами. Но есть класс задач, которые невозможно решить без программирования. Это задачи, в которых часть документа из нескольких или многих операторов надо выполнить многократно. В таких случаях документ должен состоять из отдельных подпрограмм, объединенных в единую «головную» программу.

Использование раздела «Программирование» позволяет записать в Mathcad программы любой сложности. Программирование в Mathcad чрезвычайно наглядно и понятно, так как программа представляет собой последовательность формул.

Основные операторы программирования расположены на панели Programming Toolbar (Панель программирования), вызываемой соответствующей кнопкой математической панели.

1.1 Создание программ

Программа Mathcad есть частный случай выражения Mathcad. Подобно любому выражению, программа возвращает значение, если за ней следует знак равенства. Обычное выражение Mathcad состоит из одной строки. Выражение-программа содержит много строк. Фактически это составное выражение.

Выражение-программа состоит из названия выражения, следующего за ним знака присвоения значения и необходимых выражений в правой части, записанных в столбик и объединенных слева вертикальной чертой.

1.1.1 Ввод строк в программу

Для создания программы необходимо:

- ввести имя выражения-программы;
- ввести оператор присваивания ($:=$);

- щелкнуть на кнопке панели программирования *Add Line* (Добавить строку) столько раз, сколько строк должна содержать программа;
- в появившиеся места ввода ввести нужные операторы, лишние места ввода удалить.

Чтобы создать недостающие места ввода, надо поставить синий уголок курсора в конец строки, после которой ввести новую строку. Клавишей пробела следует выделить полностью всю строку и нажать кнопку *Add Line*. При этом возможны два варианта:

- если синий уголок курсора находится в начале строки, то после нажатия *Add Line* место ввода появится выше этой строки;
- если синий уголок курсора находится в конце строки, то после нажатия *Add Line* место ввода появится ниже этой строки (см. рисунок 6.1).

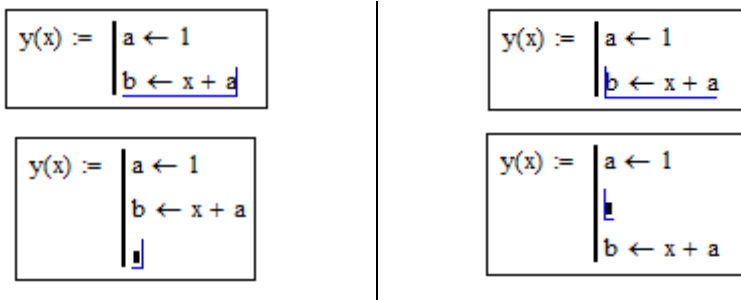



Рисунок 6.1 – Результаты ввода новой строки в программу при различном положении уголка курсора

В некоторых случаях, например, при необходимости вставить строку между двумя вложенными циклами, что при создании программы встречается очень часто, указанный прием не работает. В этом случае необходимо:

- выделить черным цветом весь внутренний цикл;
- вырезать его в буфер кнопкой *Cut* (*Вырезать*);
- нажать кнопку *Add Line*;
- вставить из буфера вырезанный фрагмент на прежнее место;
- заполнить появившееся дополнительное место ввода.

1.1.2. Локальное присвоение значений

Программа Mathcad – это обособленный модуль внутри Mathcad-документа со своими именами переменных и их значениями, которые не видны из других мест Mathcad-документа.

Присваивание значений переменным и константам в программах производится с помощью программного оператора присваивания который вводится с панели программирования нажатием кнопки  .



При создании программы, когда знаком присваивания приходится использовать часто, полезно пользоваться клавишей [{].

Локальные значения переменных нельзя использовать вне программы.



Ни оператор присвоения значения (:=), ни оператор вывода (=) в пределах программ не применяются.


Из основного документа в программу передаются в виде констант все значения величин, имеющих одинаковые имена в основном документе и в программном модуле. Эти значения сохраняются неизменными при любых действиях с ними внутри программы.

Для того чтобы любые величины стали переменными, им надо присвоить значения внутри программы с помощью программного оператора присваивания.

В большой программе, где в головную программу необходимо передать много переменных величин, всем им в начале программы необходимо присвоить значения.

Передать нужные величины в программу можно и другим способом, создав программу-функцию, где передаваемые величины записаны в параметры функции.



Оператор программирования в Mathcad не допускает наличия одинаковых имен в левой и правой частях оператора локального присвоения  . Например, массиву нельзя присвоить имя функции.

Любая программа представляет собой сочетание обычных математических выражений с операторами условия и цикла. Далее рассматривается назначение основных операторов.

1.2 Условный оператор if

Действие условного оператора состоит из двух частей. Сначала проверяется условие, записанное справа от оператора **if**. Если оно верно, выполняется выражение, стоящее слева от **if**, если не верно, происходит переход к следующей строке программы.

Чтобы вставить условный оператор в программу необходимо:

1. В создаваемой программе установить курсор на свободное место ввода, где должен появиться условный оператор.
2. На панели программирования (*Programming Toolbar*) щелкнуть на кнопке **if**. В программе появится шаблон оператора с двумя местами ввода.
3. В правое место ввода ввести условие. Следует пользоваться логическими операторами, вводя их с панели *Boolean* (*Булевы операторы*).
4. Слева от оператора **if** ввести выражение, которое должно выполняться, если условие верно. Если при выполнении условия должно выполняться сразу несколько выражений, надо иметь несколько мест ввода. Для этого следует установить курсор на место ввода слева от **if** и нажать *Add Line* столько раз, сколько строк нужно ввести.

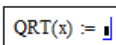
5. Если условие не верно, происходит переход к следующей строке программы. Она может содержать новое условие или быть обычным выражением.

Часто встречается условие с двумя вариантами действия: если ..., то ..., иначе Для слова «иначе» на панели *Programming* имеется оператор **otherwise** (иначе), который вводится так же, как **if**.

Пример:

Создать программу с именем «QRT», которая будет вычислять квадратный корень от введенного неотрицательного числа. Во всех остальных случаях она должна выдавать результат «0».

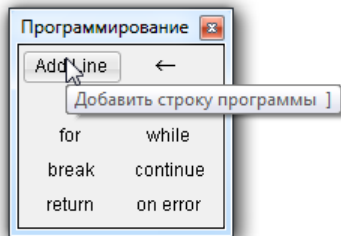
1. Ввод названия программы и её переменной



QRT(x) :=

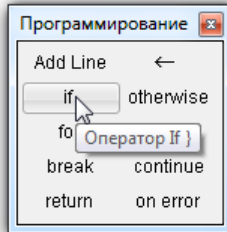
2. Нажать на панели «Программирование» кнопку *AddLine*

QRT(x) := \int



3. Ввести условный оператор **if**

QRT(x) := \int



4. Ввести условие, что переменная программы неотрицательна

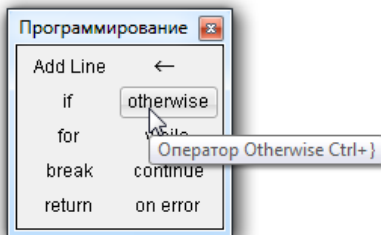
QRT(x) := \int if $x \geq 0$

5. Ввести выполняемое по условию выражение

QRT(x) := \int \sqrt{x} if $x \geq 0$

6. Ввести условный оператор **otherwise**

QRT(x) := \int \sqrt{x} if $x \geq 0$



7. Ввести выполняемое выражение

QRT(x) := \int \sqrt{x} if $x \geq 0$
 \int otherwise

1.3 Операторы цикла

Важнейшим элементом программирования, помимо условного оператора, является оператор цикла. В обычном Mathcad-документе использование дискретной переменной фактически аналогично применению оператора цикла, используемого для вычисления одного выражения. Mathcad вычисляет выражения сверху вниз и переходит к следующему выражению, лишь завершив все вычисления предыдущего выражения, поскольку вернуться к нему уже не сможет. Если же в каждом цикле должно быть вычислено несколько выражений, необходимо составлять программу.

Панель программирования Mathcad содержит два оператора цикла – операторы **for** и **while**:

- Если число выполнений цикла заранее известно, используется оператор **for**.
- Если цикл должен завершиться по выполнении некоторого условия и момент выполнения этого условия неизвестен, используется оператор **while**.

1.3.1 Оператор while

Цикл **while** выполняется до тех пор, пока остается истинным условие цикла, поэтому нет необходимости знать число вычислений заранее. Важно только, чтобы где-либо внутри цикла или в другом выполняемом участке программы присутствовал оператор, делающий условие цикла ложным. В противном случае цикл будет выполняться бесконечно.



Если выполняемая программа заиклилась, ее можно остановить, нажав клавишу Esc.

Чтобы записать цикл **while**, необходимо выполнить следующие действия:

1. Установить курсор на свободное место ввода в программе (справа от вертикальной черты).
2. На панели программирования нажать кнопку *while*. Появится шаблон с двумя местами ввода.

3. Справа от слова «while» ввести условие выполнения цикла. Обычно это логическое выражение.

4. В оставшееся поле ввода (внизу под словом «while») ввести выражение, которое вычисляется в цикле.

Если в цикле надо вычислять несколько выражений, то вначале устанавливается курсор на место ввода и нажимается кнопка *Add Line* столько раз, сколько строк будет содержать цикл. Затем заполняются все места ввода, вводя нужные выражения. Лишние места ввода при необходимости удаляются.

Обнаружив заголовок цикла **while**, Mathcad проверяет условие цикла. Если оно истинно, то Mathcad выполняет тело цикла и снова проверяет условие. Если оно ложно, Mathcad заканчивает выполнение цикла.

Пример:

Создать программу для нахождения значения и индекса первого элемента, больше чем 1,5.

| | | | | | | | |
|-------|------|------|------|------|------|------|------|
| X, мм | 1,25 | 1,28 | 1,34 | 1,45 | 1,72 | 1,81 | 1,98 |
|-------|------|------|------|------|------|------|------|

1. Ввод данных в программу

$$V := (1.25 \ 1.28 \ 1.34 \ 1.45 \ 1.72 \ 1.81 \ 1.98) \quad K := 1.5$$

2. Преобразование данных в вектор

$$V := V^T$$

3. Ввод параметров системы отсчета

$$ORIGIN := 1 \quad j := ORIGIN.length(V)$$

4. Ввод программы

$$t := \begin{cases} j \leftarrow 1 \\ \text{while } V_j \leq K \\ \quad j \leftarrow j + 1 \\ \left(\begin{array}{c} j \\ V_j \end{array} \right) \end{cases}$$

5. Результат вычислений

$$t = \begin{pmatrix} 5 \\ 1.72 \end{pmatrix}$$

1.3.2 Оператор for

В цикле **for** число повторений цикла определяется переменной, задаваемой в начале цикла. Для создания такого цикла необходимо:

1. Установить курсор на свободное место ввода в программе (справа от вертикальной черты).
2. На панели программирования нажать кнопку *for* (Цикл for). Появится шаблон с тремя местами ввода.
3. Справа от слова «for» необходимо ввести имя переменной цикла. После знака \in вводится диапазон изменения переменной цикла так же, как это делается с помощью дискретной переменной. Переменной цикла может быть ряд чисел, или вектор, или список скаляров, диапазонов, векторов, разделенных запятой.

Пример:

Создать программу для нахождения суммы значений чисел от 1 до 2 с шагом 0,2.

$$Z := \left| \begin{array}{l} S \leftarrow 1 \\ \text{for } i \in 1, 1.2.2 \\ \quad S \leftarrow S + i \\ S \end{array} \right. = 10$$

Пример:

Создать программу для объединения 2-х векторов в один

$$A := \begin{pmatrix} 15 \\ 15 \\ 17 \end{pmatrix} \quad B := \begin{pmatrix} 1 \\ 3 \end{pmatrix}$$

$$Z := \left| \begin{array}{l} m \leftarrow 1 \\ \text{for } s \in A, B \\ \quad \left| \begin{array}{l} X_m \leftarrow s \\ m \leftarrow m + 1 \end{array} \right. \\ X \end{array} \right. \quad Z = \begin{pmatrix} 0 \\ 15 \\ 15 \\ 17 \\ 1 \\ 3 \end{pmatrix}$$

1.3.3 Операторы break, continue, return

Эти операторы используются для управления работой циклов и всей программы в целом:

- **continue** возвращает расчет к началу цикла;
- **break** обеспечивает выход из цикла и продолжение работы программы;
- **return** обеспечивает выход из программы.

Пример:

Создать программу для нахождения суммы нечетных чисел от 1 до введенного числа. Если введено отрицательное число, вывести значение «Ошибка» – «Error».

```
S(n) := | "Выход из программы, если число отрицательное"  
        | return "Error" if n < 0  
        | "Обнуление начальной суммы"  
        | S ← 0  
        | "Начало цикла подсчета суммы"  
        | for i ∈ 1..n  
        | | "Переход в начало цикла, если величина i четная (делится на 2 без остатка)"  
        | | continue if mod(i, 2) = 0  
        | | S ← S + i  
        | S  
S(5) = 9  
S(-6) = "Error"
```



Для ввода комментария в программу необходимо установить курсор на свободное место ввода в программе и ввести символ " (кавычки). Появится символ введения строковой переменной: две пары кавычек с местом ввода между ними.



Для вставки в программу операторов **continue**, **break**, **return**, как и других операторов программирования, можно пользоваться только панелью программирования. Их нельзя набирать с клавиатуры.

1.4 Вывод результатов расчета из программы

Итогом действия программы является результат выполнения последнего оператора программы. Если последним оператором является математическое выражение, то результат его вычисления и будет результатом действия программы. Если необходимо выводить данные, это делается путем формирования внутри программы массивов и составных массивов.

Чтобы вывести результаты расчета, необходимо следующие действия:

1. Установить курсор на пустое место ввода последнего оператора.
2. Для вывода одного числа ввести имя соответствующей переменной (любой, которой присвоено какое-либо значение в программе или в основном документе).
3. Для вывода вектора или матрицы ввести их имя, но каждый элемент массива должен быть определен либо в программе, либо в основном документе.

Элементы вектора или матрицы можно ввести их именами. Для этого, установив курсор на место ввода последнего оператора, комбинацией клавиш **[Ctrl]+[m]** надо вызвать шаблон матрицы, ввести число строк и столбцов, ввести имена элементов, которым присвоены значения.

1.5 Подпрограммы-функции

Любая программа является выражением, состоящим из ряда выражений (объединенных вертикальной чертой), каждое из которых, в свою очередь, может быть программой. Например, если часть программы, состоящая из нескольких операторов, должна выполняться несколько раз внутри программы, ее целесообразно записать в виде подпрограммы-функции и вызывать, меняя внутренние параметры, каждый раз, когда это необходимо. Формально подпрограммой, состоящей из одного оператора, является любая функция пользователя, входящая в состав другой программы.

Как правило, подпрограмма-функция располагается где-то выше головной программы, в которой она используется. Результаты расчета можно вывести в любой позиции правее и ниже имени выражения-программы.

1.6 Отладка программ

Mathcad – перспективное средство программирования. Создать расчетную программу в Mathcad можно во много раз быстрее и значительно проще, чем на других языках программирования. Кроме того, программа Mathcad – это фактически записанный в виде последовательности формул алгоритм решения задачи. Она абсолютно понятна каждому, знающему суть задачи. В отличие от нее программы на других языках представляют собой очень трудный для понимания длинный список операторов.

Начиная с 13-ой версии Mathcad, в нем появились две функции, **Trace** и **Debug**, предназначенные для отладки программ. В Mathcad 13 разработчики предусмотрели специальную область окна Mathcad, называемую *Trace Window* (*Окно отладки*), а также дополнительную панель инструментов *Debug* (*Отладка*) и встроенные функции **trace** и **pause**. Функция **trace** осуществляет вывод наблюдаемых переменных в окно отладки; функция **pause** приостанавливает вычисления, позволяя анализировать выведенные в окне данные промежуточных вычислений.

Обращение к этим функциям:

```
trace("format string", x, y, z...);
```

```
pause("format string", x, y, z...),
```

где **format string** – строка, содержащая текст (на английском языке), предшествующий выводимому числу; **x, y, z...** – список аргументов, значения которых выводятся на экран.



Для включения этих функций в работу необходимо в меню *View* выбрать *Toolbars* → *Debug* (*Вид* → *Панели инструментов* → *Отладка*) и в появившейся на экране панели включить обе кнопки, *Toggle Debugging* и *Trace Window* (*Переключение режима отладки* и *Окно трассировки*)

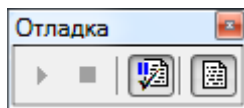


Рисунок 6.1 – Окно «Отладка»

Строка текста, взятая в кавычки, должна заканчиваться знаком (=) и порядковым номером метки в фигурных скобках, которая отмечает оператор, чей результат действия выводится в окне *Trace Window* (*Окно трассировки*).

Пример:

Провести трассировку программы для нахождения значения и индекса первого элемента, больше чем 1,5.

| | | | | | | | |
|-------|------|------|------|------|------|------|------|
| X, мм | 1,25 | 1,28 | 1,34 | 1,45 | 1,72 | 1,81 | 1,98 |
|-------|------|------|------|------|------|------|------|

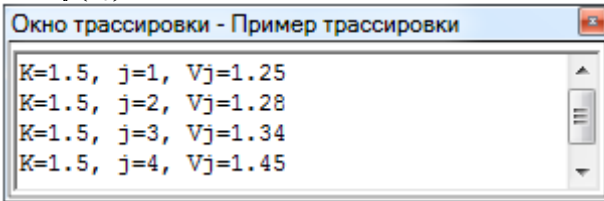
$V := (1.25 \ 1.28 \ 1.34 \ 1.45 \ 1.72 \ 1.81 \ 1.98)$ $K := 1.5$

$V := V^T$

$ORIGIN := 1$

$j := ORIGIN.length(V)$

```
t := | j ← 1
      | while Vj ≤ K
      |   | trace("K={0}, j={1}, Vj={2}K,j,Vj")
      |   | j ← j + 1
      | ( j )
      | ( Vj )
```



Для вывода результата действия функции trace выделить все выражение с этой функцией и нажать клавишу [=] или написать имя выражения-программы с функцией trace и оценить выражение, нажав клавишу [=].

1.7 Рекомендации по написанию программ

При разработке программы следует начинать со следующих действий:

1. Составить формульный алгоритм программы.
2. Выделить и записать в виде подпрограмм-функций повторяющиеся

несколько раз или просто загромождающие программу части.

3. Составить головную программу. Она должна быть сравнительно краткой и включать в себя обращения к подпрограммам.

Далее начинается отладка программы. Вначале отлаживается каждая подпрограмма. Следует просмотреть список величин, используемых в данной подпрограмме. Константы, не изменяющиеся внутри программы, должны быть заданы в основном Mathcad-документе где-либо выше места использования подпрограммы. Можно присвоить им значения и в самой подпрограмме, но нежелательно без необходимости увеличивать размер подпрограммы.



Переменным величинам, изменяющимся внутри подпрограммы, должны быть присвоены значения обязательно внутри подпрограммы, или эти величины должны быть записаны в число параметров в названии подпрограммы-функции. В противном случае величины будут иметь значения, присвоенные им в основном документе, и не смогут изменяться в процессе расчета в подпрограмме.

В конце подпрограммы следует создать место вывода результатов расчета в виде составного массива, включив в него как можно больше величин промежуточных расчетов, выполняемых в подпрограмме. Потом следует посмотреть на результаты расчета. Как правило, программа с первого раза не работает и результатов не дает. Чтобы найти ошибки нужно щелкнуть мышью внутри подпрограммы. Нужно проверить, есть ли сообщение об ошибках. Если сообщение об ошибках есть, следует исправить их, учтя сообщение.

Чаще всего появляется сообщение «*This variable or function is not defined above*» («*Эта переменная или функция еще не определена*»). В этом случае нужно проследить по документу, где этой величине должно быть присвоено значение (внутри программы, в подпрограмме, в основном документе). Если оно должно быть присвоено внутри программы, следует присвоить правильное значение, если в основном документе – проверить значение, написав имя переменной и нажав клавишу [=]. Если же значение должно быть присвоено в подпрограмме, нужно щелкнуть на ней. Если сообщений нет, необходимо проанализировать подпрограмму, как описано далее.

Часто встречается сообщение «*Value of subscript or superscript is too big (or too small) for this array*» («*Значение верхнего или нижнего индекса слишком*

велико или слишком мало для этого массива»). В этом случае, если массив вводился в основном документе, следует проверить его значения. Если массив бóльших размеров, чем был задан, значит, надо обнулить массив перед вводом значений. Если массив вводился внутри программы или подпрограммы, следует проверить диапазоны переменных циклов, их соответствие значению **ORIGIN**. Если **ORIGIN** = 1, переменные цикла не должны иметь значение 0.

Если сообщений об ошибках нет, необходимо разделить программу на части для проверки:

1. Выделить черным цветом проверяемый кусок программы и вырезать его в буфер, нажав кнопку *Cut* (*Вырезать*).

2. Щелкнуть мышью вне программы и вставить в это место из буфера вырезанный фрагмент, нажав кнопку *Paste* (*Вставить*).

3. Используемый в программировании знак присвоения ← заменить обычным оператором присвоения значения (:=).

4. Задать численные значения величин, входящих в вырезанное (и вставленное) выражение, и посмотреть результат расчета, нажав клавишу [=]. Добиться правильного ответа. Оператор цикла или условия вырезается по частям.

5. В любом месте вне программы написать имя будущей проверочной программы, оператор присваивания (:=) и, щелкнув на кнопке *Add line*, создать несколько мест ввода.

6. Выделить тело цикла или условия и вырезать его в буфер.

7. Вставить вырезанный фрагмент в любое место вне программы.

8. Выделить и вырезать в буфер заголовок цикла или условия.

9. Вставить заголовок цикла или условия в необходимое место ввода проверочной программы.

10. Выделить перечеркиванием мышью при нажатой левой кнопке вырезанное ранее тело цикла или условия и скопировать его в буфер обмена.

11. В место ввода в проверочной программе вставить из буфера тело цикла или условия.

12. Закончить составление проверочной программы, не забыть вставить составной массив вывода результатов расчета. Проверить результаты расчета.

2 Анимация

Mathcad предусматривает возможность анимации графиков и результатов вычислений путем создания AVI-файлов и вставки их в Mathcad-документ. Воспроизводятся созданные AVI-файлы встроенными в Windows средствами.

Mathcad имеет встроенную переменную **FRAME**, предназначенную только для создания анимационных клипов. **FRAME** принимает целочисленные значения от 0 до n и должна быть включена в расчетные выражения так, чтобы с изменением **FRAME** изменялся вид соответствующего графика. Фактически **FRAME** – это номер кадра.

Переменной **FRAME** нельзя присваивать какое-либо значение в Mathcad-документе. Диапазон значений **FRAME** задается в диалоговом окне главного меню командой *Tools* → *Animation* → *Record* (*Инструменты* → *Анимация* → *Запись*). Перед созданием анимационного клипа следует убедиться, что в главном меню Mathcad в окне *Tools* включен режим автоматических вычислений.

2.1 Создание анимации

Процесс создания анимации состоит из следующих действий:

1. Ввести переменную **FRAME** в математическое выражение, определяющее вид графика.
2. Выбрать пункт меню *Tools* → *Animation* → *Record* (*Инструменты* → *Анимация* → *Запись*).
3. После открытия диалогового окна анимации курсором выделить нужный фрагмент изображения.
4. В диалоговом окне задать общее число кадров и частоту их воспроизведения. Установленное по умолчанию число 10 – слишком большая скорость воспроизведения кадров. Лучше установить 2-3 кадра в секунду, иначе анимация будет длиться не более секунды.
5. Щелкнуть на кнопке *Animate* (*Анимировать*) и наблюдать за созданием кадров. Затем появится окно проигрывателя видеофайлов. В этом окне можно установить размер кадра и скорость воспроизведения. В появившемся окне

Playback можно выбрать *Вид* (*Половинный размер*, *Обычный размер* – в масштабе 1:1, *Двойной размер*) и скорость воспроизведения кадров. Если в окне анимации оставлено по-умолчанию 10 кадров в секунду, то движком следует уменьшить скорость примерно до 30 %. Если в окне анимации установлено 3 кадра в секунду, то регулятор скорости не стоит трогать, а то скорость движения станет маленькой и эффекта анимации не получится. Размеры окна можно изменить и обычным способом, то есть растянув его за угол мышью. Однако при этом пропорции окна могут измениться.

6. Созданный клип сохранить с помощью команды *Save As* (*Сохранить как*). Сохраненный файл можно использовать и вне Mathcad-документа.

7. Если сохраненный как AVI-файл клип необходимо вставить в Mathcad-документ, то для этого в главном меню Mathcad нужно выбрать команду *Insert* → *Object* (*Вставка* → *Объект*). Откроется окно *Вставка объекта*. В этом окне необходимо выбрать пункт *Создать из файла*. Далее следует выбрать нужный файл, щелкнув в диалоговом окне на кнопке *Обзор*.

Желательно нажать кнопку *Связь* в том же окне. Это позволит редактировать AVI-файл непосредственно из Mathcad-документа.

Если в том же окне установить флажок *В виде значка* и нажать *ОК*, то на месте курсора в Mathcad-документе появится значок AVI-файла. Двойной щелчок мышью на этом значке вызовет появление проигрывателя видеофайлов и исполнение анимации.

Если флажок *В виде значка* не устанавливать, то после нажатия кнопки *ОК* на месте курсора в Mathcad-документе появится картинка с первым кадром AVI- файла. Двойной щелчок мышью на этом значке вызовет появление проигрывателя видеофайлов и исполнение анимации.

Вставленный в документ AVI-файл можно отредактировать. Для этого необходимо щелкнуть на объекте (значке или картинке) правой кнопкой мыши. В появившемся контекстном меню выбрать пункт *Связанный объект: Видеозапись* → *Правка*.

В открывшемся окне редактирования следует выбрать пункт *Правка* → *Параметры*. Желательно установить флажок *Автоповтор*, тогда анимация будет длиться бесконечно долго, до тех пор, пока не щелкнуть мышью вне объекта.

Файлы анимации (AVI-файлы) обычно занимают много места на диске. Mathcad позволяет сохранять их в сжатом формате. Перед созданием анимации можно выбрать, какие методы сжатия использовать и сжимать ли файл вообще. Для этого в диалоговом окне *Animate* необходимо выбрать *Options (Свойства)*, а в окне *Compressor Options (Параметры сжатия)* – программу сжатия или команду *Полные кадры* (без сжатия).



При создании анимационных клипов рекомендуется отключить автоматическое масштабирование графика, так как возможны скачки изображения при изменении масштаба. Следует ввести числовые значения на осях графика вручную.

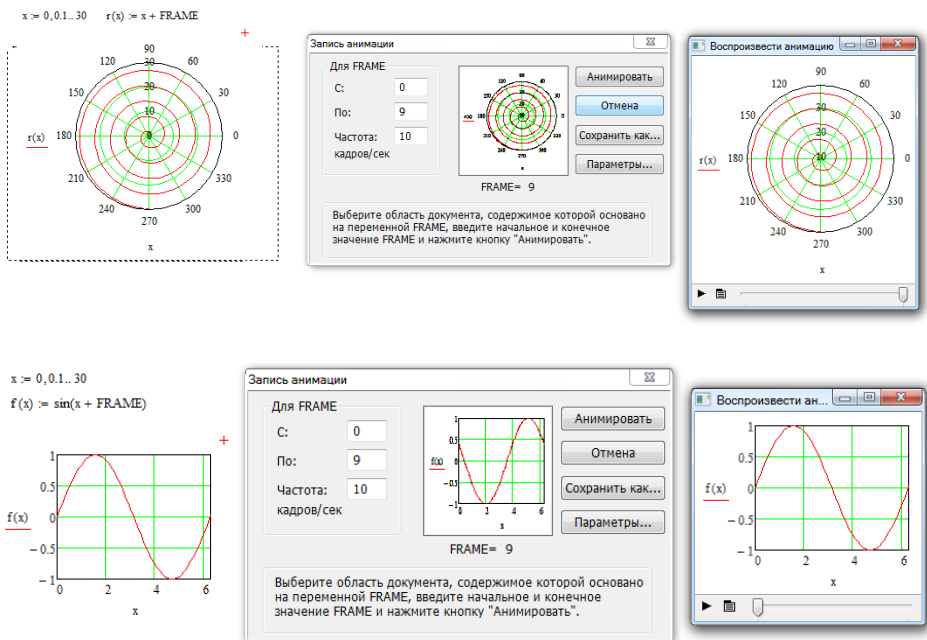


Рисунок 6.2 – Примеры анимации

ПРИЛОЖЕНИЯ

Приложение 1. Горячие клавиши Mathcad

1 Операторы

Приведённые ниже комбинации клавиш, используются для вставления операторов. За единственным исключением (клавишей кавычки ["]) результат нажатия этих клавиш одинаков в математической области и в пустом месте.

| Клавиша (и) | Назначение |
|--------------|--|
| ! | Факториал. |
| “ | В математической области создаёт оператор комплексного сопряжения. |
| “ | В пустом месте создаёт текстовую область. |
| # | Произведение по дискретному аргументу. |
| \$ | Суммирование по дискретному аргументу. |
| & | Интеграл. |
| ‘ | Пара скобок. |
| , | Разделяет выражения, откладываемые по одной оси графика. |
| ; | Предшествует второму числу при задании диапазона. |
| : | Предшествует последнему числу при задании диапазона. |
| ? | Производная первого порядка. |
| [| Нижний индекс. |
| \ | Квадратный корень |
| ^ | Показатель степени. |
| | Абсолютное значение. |
| Ctrl+1 | Транспонирование. |
| Ctrl+3 | Не равно. |
| Ctrl+4 | Сумма элементов вектора. |
| Ctrl+8 | Векторное произведение. |
| Ctrl+9 | Меньше либо равно. |
| Ctrl+0 | Больше либо равно. |
| Ctrl+Shift+4 | Суммирование по индексу. |
| Ctrl+Shift+3 | Произведение по индексу. |
| Ctrl+Shift+? | Производная n-ного порядка. |
| Ctrl+\ | Корень n-ной степени. |
| Ctrl+Enter | Сложение с переносом. |

2 Окна и рабочие документы

Следующие комбинации клавиш используются для манипуляции окнами и рабочими документами как целым.

| Клавиша (и) | Назначение |
|-------------|---|
| Alt+Esc | Переключиться на другую задачу Windows. |
| Alt+Tab | Переключиться на перечень задач Windows |
| Ctrl+F4 | Закреть рабочий документ. |
| Ctrl+F6 | Активизировать следующее окно. |
| Alt+F4 | Выйти из Mathcad. |
| Ctrl+R | Обновить экран. |
| F1 | Открыть окно Справки. |
| F5 | Открыть рабочий документ. |
| F6 | Сохранить рабочий документ. |
| F7 | Создать рабочий документ. |
| F9 | Пересчитать результаты на экране. |
| Shift+F1 | Включить контекстную Справку. |

3 Редактирование

Следующие комбинации клавиш используются для редактирования документов Mathcad.

| Клавиша (и) | Назначение |
|-------------|---|
| Ctrl+F9 | Вставляет чистую строку. |
| Ctrl+F10 | Удаляет чистую строку. |
| Ctrl+F5 | Вызывает диалоговое окно для поиска фрагмента строки. |
| Shift+F5 | Вызывает диалоговое окно для поиска и замены фрагмента строки |
| Alt+BkSp | Отменяет последнее действие по редактированию документа. |
| Ctrl+C | Копирует выделение в буфер обмена. |
| Ctrl+V | Вставляет в документ содержимое буфера обмена. |
| Ctrl+X | Вырезает выделение и помещает его в буфер обмена. |
| Ctrl+U | Вызывает диалоговое окно для вставки встроенных единиц измерения. |
| Ins | Переключает между режимами редактирования вставка и забивка |

Приложение 2. Сообщения об ошибках

Это приложение является алфавитным списком диагностических сообщений об ошибках в математических выражениях. Они появляются при попытке ввода, обработки или вычисления выражения, в котором Mathcad обнаруживает ошибку.

Если Mathcad находит ошибку при попытке вычисления функции, определенной пользователем, он помечает сообщением об ошибке имя функции, а не ее определение. В этом случае проверьте определение функции, чтобы понять, что вызвало ошибку.

| Сообщение | | Причина |
|----------------------------|--------------------------------|---|
| Англ. яз. | Рус. яз. | |
| Array size mismatch | Несовпадение размеров массивов | Попытка произвести операцию с векторами или матрицами, размеры которых не подходят для этой операции. Многие операции требуют, чтобы их векторные аргументы были одного размера, например произведение или функции linterp и corr . |
| Cannot be denned | Не может быть определено | Слева от символа определения (:=) помещено неопределяемое выражение. |
| Cannot take subscript | Не содержит верхних индексов | Верхний индекс использован не для матрицы, а для чего-то другого. |
| Cannot take subscript | Не содержит нижних индексов | Нижний индекс использован не для вектора или матрицы, а для чего-то другого. |
| Definition stack over- How | Переполнение стека определении | Использовано слишком много вложенных функций. |
| Did not find solution | Решение не найдено | Mathcad не нашел решения системы уравнений. Чтобы блок решения уравнений выдал в качестве решения приближающийся результат, используйте функцию Minerr вместо функции Find . |

| | | |
|-------------------------|------------------------------|---|
| Domain error | Ошибка области определения | Попытка вычислить значение функции, имеющей аргумент, выходящий за область определения |
| Duplicate | Дублирование | Попытка определить одну переменную дважды в одном определении. Это сообщение появляется, когда создается вектор по левую сторону определения и используете одно имя в этом векторе дважды. |
| Equation too large | Слишком большое выражение | Для вычисления в Mathcad введено слишком большое выражение. Разделите выражение на два или более подвыражений. |
| Error in constant | Ошибка в константе | Mathcad интерпретирует указанное выражение как некорректную константу. Mathcad воспринимает все начинающееся с цифры как константу. |
| Error in list | Ошибка в списке | Указанная функция содержит некорректный список аргументов. Правильное определение функции начинается таким образом: $f(x, y, z \dots)$. Список аргументов в скобках может состоять из одного или нескольких имен, разделенных запятой. |
| Error in solve block | Ошибка в блоке | Сообщение при вычислении пользовательской функции, выражаемой через блок решения уравнений, содержащий ошибку. |
| File error | Ошибка файла | Система столкнулась с ошибкой при чтении файла с помощью функции READ или READPRN . |
| File not found | Файл не найден | Система не нашла файла данных, указанного в качестве параметра для функции READ или READPRN , либо для импорта в графическую область. |
| Illegal array operation | Неверная операция с массивом | Попытка применить к вектору или матрице функцию или оператор, которые требуют скалярные аргументы. Например, это сообщение можно видеть при попытке применения функции синус к квадратному корню из матрицы. |
| Illegal context | Неверный контекст | Оператор или функция использованы в контексте, запрещаемом Mathcad. |

| | | |
|-----------------------|-------------------------------------|---|
| Illegal factor | Неверный множитель | В поле ввода единиц в конце выражения, возвращающего численный результат, введено неверное выражение. Допустимы вещественные ненулевые скалярные значения. |
| Illegal function name | Неверное имя функции | Использовано выражение, которое Mathcad интерпретирует как функцию, но имя функции неверно. Это сообщение появится, например, в случае использования числа как имени функции: $6(x)$. Чаще всего оно возникает, если пропущен оператор типа $*$, что заставляет Mathcad интерпретировать скобки в выражении как признак функции, а не как группирование операций. |
| Illegal ORIGIN | Неверное употребление | ORIGIN определен через нецелое значение или значение с величиной, большей 16 000 000. Это сообщение отмечает первое использование индекса после неверного употребления ORIGIN . |
| Illegal range | Неправильный диапазон | Дискретный аргумент определен неправильно. |
| Illegal tolerance | Некорректная точность аппроксимации | Сообщение отмечает выражение, использующее TOL интеграл, или вхождения Root , Find или Minerr , для которых $TOL > 1$ или $TOL < 0$. Для устранения этой ошибки нужно где-либо выше отмеченного выражения установить значение TOL между нулем и единицей. |
| Incompatible units | Несовместимые единицы | Отмечает выражение, в котором складываются, вычитаются или выполняются иные операции с выражениями, имеющими различную размерность. |
| Index out of bounds | Индекс вне границ | Сообщение помечает индекс, ссылающийся на несуществующее значение массива. Такое сообщение можно видеть при использовании отрицательного верхнего или нижнего индекса (или индекса, меньшего, чем ORIGIN , если ORIGIN > 0) либо при использовании верхнего или нижнего индекса для ссылки на элемент массива с номером, большим, чем возможно согласно определению в документе. |
| Invalid order | Неверный порядок | Отмечает попытку вычислить производную с указанным порядком, который не является целым числом от 0 до 5 включительно. |

| | | |
|-----------------------|---------------------------------|--|
| List too long | Длинный входной список | Введено слишком много элементов в списке, разделенном запятыми. Это может произойти при попытке вывести на график больше выражений, чем допускается Mathcad, или при попытке создать таблицу с более чем пятьюдесятью элементами. |
| Missing operand | Пропущенный операнд | В выражении пропущен один из операндов. Например, это сообщение можно видеть при вводе знака плюс без ввода слагаемых и последующем нажатии знака равенства. Mathcad показывает поле ввода (маленький прямоугольник) на месте пропущенного операнда, |
| Misplaced comma | Неуместная запятая | Запятая использована там, где ее не должно быть. |
| Missing operand | Пропущенный операнд | В выражении пропущен один из операндов. Например, это сообщение можно видеть при вводе знака плюс без ввода слагаемых и последующем нажатии знака равенства. Mathcad показывает поле ввода (маленький прямоугольник) на месте пропущенного операнда, |
| Missing operator | Пропущенный знак операции | В выражении или уравнении пропущен один из знаков операции. |
| Must be 3-vector | Должно быть трехмерным вектором | Попытка найти векторное произведение от операндов, не являющихся трехмерными векторами. Векторное произведение определено только для векторов с тремя элементами. |
| Must be array | Должно быть массивом | Попытка выполнить операцию, которую можно выполнять только на массиве, со скаляром. |
| Must be dimensionless | Должно быть безразмерным | Указанное выражение имеет размерность, хотя ситуация требует, чтобы оно было безразмерным. Единицы измерения нельзя использовать для аргументов некоторых функций (например, cos и In) или в показателе степени. |
| Must be increasing | Должно быть возрастающим | Вектор, элементы которого не расположены в порядке строгого возрастания, использован в качестве аргумента одной из функций (spline , pspline , cspline , interp , linterp и hist). Первый аргумент этих функций должен быть вектором со строго возрастающими элементами. |

| | | |
|------------------------|-----------------------------------|---|
| Must be integer | Должно быть целым | Использовано нецелое выражение там, где требуется целое, например как аргумент функции identity или как индекс, нижний или верхний. |
| Must be nonzero | Должно быть ненулевым | Попытка вычислить встроенную функцию от нуля, хотя для нуля она не определена. |
| Must be positive | Должно быть положительным | Сообщение отмечает чертеж, в котором одна из границ по оси, использующей логарифмический масштаб, равна нулю или отрицательна. Mathcad может выводить на график вдоль логарифмической оси только положительные значения. |
| Must be range | Должен быть диапазон | Использовано, что-либо, не являющееся дискретным аргументом. |
| Must be real | Должно быть вещественным | Мнимое или комплекснозначное выражение использовано там, где Mathcad требует вещественнозначное выражение. |
| Must be scalar | Должно быть скаляром | Векторное или матричное выражение использовано там, где требуется скаляр, например в качестве аргумента функции identity . |
| Must be vector | Должно быть вектором | Сообщение отмечает скаляр или матрицу в операции, требующей векторный аргумент. |
| No matching Given | Нет соответствующего Given | Сообщение указывает на функции Find или Minerr без соответствующего им слова Given . |
| No scalar value | Нескалярная величина | Вектор или выражение, содержащее дискретный аргумент, используются там, где требуется скалярная величина. |
| Not a name | Не является именем | Число или другая комбинация символов использованы там, где Mathcad требует имя, например как второй аргумент функции root . |
| Only one array allowed | Допустим только один массив | Попытка ввести более чем один массив в поле ввода для карты линий уровня. Mathcad в этом случае допускает не более чем один массив, поскольку карта линий уровня может представлять не более чем одну функцию одновременно. |
| Overflow | Переполнение | Попытка вычислить выражение, которое превосходит наибольшее число, которое может быть представлено Mathcad (примерно 10). |

| | | |
|-----------------------|-------------------------|--|
| Significance lost | Потеряны значащие цифры | Сообщение отмечает попытку вывести функцию от величины, которая лежит за пределами диапазона, где значение функции может быть вычислено точно. |
| Too many points | Слишком много точек | Попытка вывести на график точек больше, чем Mathcad может обработать для одного графика. |
| Too many subscripts | Слишком много индексов | Использовано два или более нижних индекса для вектора либо три или более индекса для матрицы. |
| Undefinet | Не определено | Показанное в негативном изображении имя функции или переменной не определено. Чтобы его определить, введите имя переменной с последующим двоеточием (:) и выражение или число, ее определяющее. Это сообщение часто означает, что для определения переменной использован знак равенства (=) вместо двоеточия. |
| Unmatched parenthesis | Дисбаланс скобок | Попытка вычислить выражение, содержащее левую скобку без соответствующей ей правой. Исправьте выражение, удалив левую скобку или поставив в нужном месте правую. |
| Wrong size vector | Неверный размер вектора | Сообщение указывает на функцию преобразования Фурье, аргумент которой имеет число элементов, отличное от допустимого, fft требует в качестве аргумента вектор с количеством элементов 2^n , где n - целое число, большее 1. ifft требует вектор с $1+2^n$ элементами, где n - целое число, большее 0. Если ORIGIN равен нулю, Mathcad автоматически включает элемент с нулевым индексом как компоненту вектора-аргумента. |

Список литературы

1. Mathcad (+ CD-ROM): Евгений Макаров – Санкт–Петербург, Питер, 2009 г.– 384 с.
2. Mathcad 11/12/13 в математике. Справочник (+ CD-ROM): В. П. Дьяконов – Москва, Горячая Линия – Телеком, 2007 г.– 960 с.
3. Mathcad 12. Для студентов и инженеров: Валерий Очков – Москва, БХВ–Петербург, 2005 г. – 464 с.
4. Mathcad 12. Наиболее полное руководство (+ CD-ROM): Дмитрий Кирьянов – Москва, БХВ–Петербург, 2005 г.– 566 с.
5. Mathcad 12: Е. Р. Алексеев, О. В. Чеснокова – Санкт–Петербург, НТ Пресс, 2005 г.– 352 с.
6. Mathcad 14 для студентов и инженеров. Русская версия: Валерий Очков – Санкт–Петербург, БХВ–Петербург, 2009 г.– 512 с.
7. Автоматизация инженерных расчетов в среде Mathcad: Щепетов А.Г. – Москва, 2006 г.– 264 с.
8. Прикладные математические пакеты. Часть 1. MathCAD: Б. А. Акишин, Н. Х. Эркенов – Санкт–Петербург, РадиоСофт, 2009 г. – 132 с.
9. Решение задач вычислительной математики в пакетах Mathcad 12, MATLAB 7, Maple 9: Е. Р. Алексеев, О. В. Чеснокова – Москва, НТ Пресс, 2006 г.– 496 с.
10. Соппротивление материалов на базе Mathcad (+ CD-ROM): Е. Г. Макаров – Москва, БХВ–Петербург, 2004 г. – 512 с.
11. Теоретическая механика на базе Mathcad. Практикум: В. Д. Бергтяев – Москва, БХВ–Петербург, 2005 г.– 752 с.