ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Решетникова Г.Н., Хлебников А.А., Арцер П.А., Гриднев А.А.

MathCAD PLUS 6.0 PRO

Учебное пособие

Томск 2000

В настоящем учебном пособии приведены краткие сведения об основных приемах работы в системе MathCAD PLUS 6.0 PRO, предназначенной для выполнения математических расчетов различной степени сложности. Достаточно полно рассмотрены графические возможности, программирование и символьная математика системы MathCAD. Описаны правила работы с документами. Рассмотрены функции, предназначенные для обработки наборов данных, решения линейных и нелинейных уравнений и систем, обыкновенных дифференциальных уравнений и систем и дифференциальных уравнений в частных производных. Приведено краткое описание матричных операторов и функций, статистических функций и датчиков случайных чисел. Изложение сопровождается примерами, иллюстрирующими технологию работы в системе MathCAD.

Учебное пособие предназначено для студентов и преподавателей вузов и других образовательных учреждений.

Введение

Система MathCAD разработана фирмой MathSoft, Inc. (USA). Первая версия MathCAD появилась в 1986 г. Она работала под управлением 16-битной операционной системы MS-DOS и по своим возможностям недалеко уходила от современного инженерного калькулятора.

С переводом MathCAD в начале 90-х годов на платформу Windows 3.х, система стала 32-битной (с использованием Win32s для Windows 3.х), а также получила все возможности программного интерфейса приложений WinAPI: настройку под любые типы видеокарт, принтеров, использование различных шрифтов, редактирование большого количества (до 80-и) файлов MathCAD одновременно, возможности OLE (Object Linking and Embedding - связывание и внедрение объектов) и т.д.

Начиная с версии 3.0, MathCAD стал системой компьютерной алгебры, так как появилась возможность выполнения символьных (аналитических) расчетов. Для этого в MathCAD было введено урезанное ядро символьных операций системы MAPLE V.

Версия MathCAD 6.0 PRO обогатилась возможностью создания и использования программ. Несмотря на то, что элементов программирования мало (всего семь), с их помощью можно создавать достаточно сложные если программы, особенно использовать многомодульную технологию программирования. Версия MathCAD 6.0 PRO характеризуется эффективными средствами оформления документов, создания анимационных графиков и сопровождения, богатым набором шрифтов, возможностью ЗВУКОВОГО документов Windows, всех прекрасной использования графикой И современным многооконным интерфейсом.

Система MathCAD существенно отличается от аналогичных прикладных систем, таких как MatLAb, Mathematica, Maple и др. тем, что MathCAD является единственной прикладной системой, в которой описания математических задач и их решений задаются с помощью обычных в математике символов, формул и операторов. Таким образом, возможности системы MathCAD вполне оправдывают аббревиатуру CAD (Computer Aided Design - автоматизированное проектирование). А объединение текстового, формульного и графического редакторов делает эту систему универсальной.

Настоящее учебное пособие написано коллективом авторов под руководством доцента факультета прикладной математики и кибернетики ТГУ Г.Н. Решетниковой на основе опыта проведения учебных занятий в ТГУ и ТУСУРе по численным методам, моделированию систем управления, пакетам прикладных программ и другим курсам. Оно ориентировано на англоязычную версию системы MathCAD PLUS 6.0 PRO как на наиболее распространенную, при этом сопровождение английского текста русским переводом значительно

3

облегчает работу с системой. Пособие отличается компактностью изложения и сокращенным описанием тех элементов интерфейса, которые известны каждому пользователю, имеющему дело с Windows. Изложение и компоновка материала позволяют использовать пособие в качестве справочного руководства для широкого круга пользователей, занимающихся различными математическими расчетами.

Приведенный в конце перечень ошибок и комментариев к ним значительно облегчает работу непосредственно в системе.

1. Начальные сведения о системе MathCAD

1.1. Интерфейс системы MathCAD

Под интерфейсом пользователя подразумевается совокупность средств графической оболочки MathCAD, обеспечивающих удобное управление системой как с помощью клавиатуры, так и с помощью мыши. Интерфейс системы создан таким образом, что пользователь, имеющий элементарные навыки работы с Windows-приложениями, может сразу начать работу с MathCAD.

В верхней части окна находится несколько строк с основными элементами интерфейса. Верхняя строка является *строкой заголовка*. Она отображает название загруженного или вводимого с клавиатуры документа и его порядковый номер. В левой части строки заголовка помещена стандартная кнопка управления окном, а в правой части — кнопки свертывания окна в пиктограмму и развертывания его во весь экран. Одновременно можно открыть не более 8 документов MathCAD.

Вторая строка окна системы содержит опции главного меню:

<u>F</u> ile (Файл)	- работа с файлами, сетью Internet и E-mail;
<u>E</u> dit (Правка)	– редактирование документов;
<u>Т</u> ехt (Текст)	 работа с текстовым редактором;
<u>M</u> ath (Математика)	 управление процессом вычислений;
<u>G</u> raphics (Графика)	 работа с графическим редактором;
<u>S</u> ymbolic (Символика) – выбор операций символьного процессора;
<u>W</u> indow (Окно)	– управление окнами системы;
<u>B</u> ooks (Книги)	– работа с электронными книгами;
<u>H</u> elp (Справка)	 – работа со справочной базой данных о системе.

В третьей строке окна системы находятся *пиктограммы* вывода *наборных панелей*. Наборные панели в окне редактирования появляются при активизации соответствующих пиктограмм. Большинство кнопок на наборных панелях вводят общепринятые и специальные математические знаки и операторы, выводя их шаблоны в место, указанное курсором (см. рис. 1.1). Любую наборную панель можно переместить в удобное место экрана, уцепившись за ее верхнюю часть мышью.

Четвертую строку окна системы занимает *панель инструментов* (Toolbox). Окно содержит несколько групп пиктограмм управления, каждая из которых дублирует одну из важнейших команд главного меню.



Рис. 1.1. Наборные панели и соответствующие им пиктограммы

Пиктограммы операций с файлами

- Ľ Создание нового документа с очисткой окна редактирования.
- Ē Загрузка ранее созданного документа из диалогового окна.
- Запись текущего документа с его именем.
- Распечатка документа на принтере.

Пиктограммы операций редактирования

1 X

Отмена предыдущей операции редактирования.

- Перенос с очисткой выделенной части документа в буфер обмена.
- **B** Копирование выделенной части документа в буфер.
- 圇 Перенос содержимого буфера обмена в окно редактирования.

Пиктограммы размещения блоков

- Выравнивание блоков по горизонтали.
- Выравнивание блоков по вертикали.

Пиктограммы текстовых операций

- А Создание текстовой области.
- Создание текстового параграфа.
- АВС Проверка орфографии (только для англоязычного текста).

Пиктограммы установки режимов работы

- Включение режима автоматических вычислений.
- Включение режима Live Symbolics для символьных вычислений.

Пиктограммы операций с выражениями

- Вычисление выделенного выражения.
- 🖻 Вставка функции из списка.

Пиктограммы управления окном

- Ð. F
- Управление размером (масштабом) экрана.
- Вставка единиц измерения физических величин.

Пиктограммы работы со справочной базой данных



- Быстрый вызов примеров применения системы.
- Вызов справочной базы данных о системе.

Необходимо отметить, что строки с пиктограммами тоже можно свернуть мышью в наборную панель и поместить в любое место экрана. Так интерфейс системы модифицируется, и пользователь может подстроить его под свой вкус.

Пятая строка верхней части экрана содержит типовые *средства* управления шрифтами: переключатели типа символов, набора фонтов шрифта и размеров шрифта, три пиктограммы типа шрифтов (полужирный, курсив и подчеркивание), а также три пиктограммы расположения символов в строке (рядом, надстрочное и подстрочное).

В нижней части экрана кроме полосы горизонтальной прокрутки расположена еще одна строка - *строка состояния*. В ней выводятся служебная информация, краткие комментарии, номер страницы документа и др. Эта информация полезна для оперативной оценки состояния системы в ходе работы с нею.

1.2. Подменю <u>Math</u> главного меню

В опции <u>Math</u> (Математика) главного меню сосредоточены команды управления вычислительными процессами:

Matrices...(Матрицы... [Ctrl]+М) – создать матрицу или вектор;

<u>Built-In Variables...(Встроенные переменные...)</u> – установить значения встроенных переменных;

<u>Units (Единицы)</u> – выводит подменю:

In<u>s</u>ert Unit...(Вставить единицы... [Ctrl]+U) – вставить единицы измерений;

<u>Change System of Units...(Сменить систему единиц...)</u> – изменить систему единиц измерений;

<u>Dimensional Format...(Формат размерности...)</u> – изменить названия основных единиц измерений;

Choose Function...(Вставить функцию...) – показать список функций;

<u>Randomize...(Генератор случайных чисел...)</u> – установить базу генератора случайных чисел;

Calculate (Пересчитать [F9]) – произвести расчеты и обновить документ в пределах экрана;

<u>Calculate Worksheet (Пересчитать все)</u> – произвести расчеты и обновить весь документ;

<u>Toggle Equation (Отключить выражение)</u> – блокировать или разблокировать вычисление выражения;

Highlight Equation (Подсветить выражение) – изменить цвет выделенного выражения;

<u>Automatic Mode (Автоматический режим)</u> – включить или выключить автоматический режим вычислений;

Live Symbolics (Символика) – включить или выключить режим символьных вычислений;

<u>Optimize</u> (Оптимизация) – включить или выключить режим оптимизации численных расчетов;

Numerical <u>Format...(Формат числа...)</u> – изменить формат вывода чисел;

<u>Font Tag...(Шрифтовая бирка...)</u> – изменить свойства шрифта выделенного имени или всех имен, имеющих ту же шрифтовую бирку;

Change to <u>Greek Variable</u> (Греческий алфавит [Ctrl]+G) – заменить выделенную букву на греческую.

1.3. Встроенные переменные

В MathCAD есть переменные, значения которых определены сразу после запуска программы. Эти переменные называются *предопределенными или встроенными переменными*. Предопределенные переменные или имеют общепринятое значение, или используются как внутренние переменные, управляющие работой MathCAD:

- $\pi = 3.141592653589793$ число π . В численных расчетах MathCAD использует значение π с учетом 15 значащих цифр. В символьных вычислениях π сохраняет своё точное значение. Вводится с помощью наборных панелей или нажатием клавиш [Ctrl] +P;
- e = 2.718281828459045 основание натуральных логарифмов. В численных расчетах MathCAD использует значение числа е с учётом 15 значащих цифр. В символьных вычислениях е сохраняет своё точное значение;
- ∞ = 10³⁰⁷ бесконечность. В численных расчетах это конечное число. В символьных вычислениях ∞ сохраняет свое точное значение. Вводится с помощью наборных панелей или нажатием клавиш [Ctrl] +Z;
 % = 0.01 процент

		npodenn.		
TOL = 0.001	—	погрешность численных расчетов;		
$\mathbf{ORIGIN} = 0$	—	индекс первого элемента массива;		
PRNCOLWIDTH = 8	—	ширина столбца, используемая при записи файлов		
		функцией WRITEPRN;		
PRNPRECISION = 4	—	число значащих цифр, используемых при записи		
		файлов функцией WRITEPRN;		
$\mathbf{FRAME} = 0$	—	используется для создания и просмотра анимации.		
		Когда анимации не используются, значение		

FRAME равно нулю.

Хотя эти переменные уже имеют значения при запуске MathCAD, их можно переопределять. Например, если нужно использовать переменную, называемую **e**, со значением иным, чем используемое MathCAD, следует ввести новое определение, например **e**:=2. Переменная **e** примет в рабочем документе новое значение всюду ниже этого определения. Предопределенные переменные MathCAD определены для шрифтов всех гарнитур, размеров и начертаний. Это означает, что если переопределить переменную **e**, набранную одним шрифтом, как показано выше, то значение переменной **e**, набранной другим шрифтом, не изменится.



Рис. 1.2. Окно встроенных переменных MathCAD

Можно управлять значениями **TOL**, ORIGIN, PRNPRECISION И **PRNCOLWIDTH**, не вставляя их явных определений в рабочий документ. Для этого нужно выбрать команду Built-In Variables (Встроенные переменные) из (Математика) опции Math главного меню, и появится диалоговое окно (см. рис. 1.2). Чтобы установить новое значение любой из этих переменных, нужно ввести его в соответствующее поле и нажать ОК.

Далее необходимо выбрать команду <u>Calculate Worksheet</u> (Пересчитать всё) из опции <u>Math</u> (Математика), чтобы новое значение встроенной переменной было учтено при пересчете существующих формул. Числа в скобках справа от имён переменных указывают значения по умолчанию для этих переменных. Справа от полей указаны диапазоны допустимых значений соответствующих переменных.

1.4. Установка формата вывода чисел

Команда Numerical <u>Format</u> (Формат числа) из опции <u>Math</u> (Математика) главного меню отображает диалоговое окно (см. рис. 1.3), сообщающее о *формате вывода* числовых данных системы. Это окно содержит





три группы контролей. В первой устанавливается основание системы Radix: Decimal счисления (Десятичный), Нех (Шестнадцатеричный) или Octal (Восьмеричный). Bo второй, Imaginary (Мнимая единица), задается символ мнимой единицы для комплексных чисел: і или ј. Третья группа, Precision (Точность), задает точность вывода чисел. Задаются: число отображаемых знаков после запятой Displayed Precision (Выводимая точность), границы представления чисел в экспоненциальной форме Exponential Threshold (Диапазон показателя). допустимая граница для комплексных чисел Complex Tolerance (Комплексная точность) и допустимая граница для

действительных чисел Zero Tolerance (Точность нуля). Если Re (Z) / Im (Z) >10ⁿ, то комплексное число Z выводится как действительное, a если $Im(Z)/Re(Z) > 10^{n}$, то число Z выводится как мнимое. Значение n задает параметр Complex Tolerance. Если действительное число больше 10^n , то оно представляется в экспоненциальной форме. Здесь n – значение параметра Exponential Threshold. Если значения чисел по модулю меньше указанных параметром Zero Tolerance, то числа представляются в виде нулей. Внутреннее представление чисел MathCAD всегда имеет максимальную точность. С помощью диалогового окна можно выбрать формат Global (глобальный) или Local (локальный). Глобальный формат задает представление для всех числовых данных документа, а локальный – только для блока, к которому применялась операция изменения формата.

1.5. Алфавит, константы, переменные, массивы, дискретные аргументы

Прежде всего, MathCAD требует от пользователя корректное описание алгоритма решения математической задачи на входном языке, очень напоминающем общепринятый язык описания математических и научно-технических расчетов. Естественно, это описание должно быть исчерпывающе полным и абсолютно точным. Тем не менее, сказанное не означает, что решение задач в системе MathCAD нельзя рассматривать как программирование. Просто MathCAD обладает специализированным входным языком программирования очень высокого уровня, ориентированным на математические расчеты. Поэтому, рассматривая входной язык системы как язык программирования, можно выделить в нем типичные для любого языка программирования понятия и объекты, такие как идентификаторы, константы, переменные, массивы и другие типы данных, операторы и функции, управляющие структуры и т. д.

Алфавит системы MathCAD содержит прописные и строчные буквы латинского, греческого алфавитов, арабские цифры от 0 до 9, системные переменные, операторы, имена встроенных функций, специальные знаки.

В системе MathCAD используются *числовые константы*, значения которых представляют собой числа определенного типа: десятичные (123, -12.3, $12.3 \cdot 10^{-5}$), диапазон их изменения от 10^{-307} до 10^{307} ; комплексные (2-1i, 3i); восьмеричные и шестнадцатеричные.

МаthCAD может работать и с *размерными величинами*. Помимо своего числового значения они характеризуются еще и указанием на то, к какой физической величине они относятся. В MathCAD используется два вида единиц измерения: первичные (или основные) и производные – и три системы единиц измерения: MKS, U.S. (американская) и CGS. К первичным единицам относятся: **М** (масса), **L** (длина), **T** (время), **Q** (заряд), **K** (температура). Первичные единицы **всегда** вводятся непосредственно за числом, например,

4.50, ЗМ. Производные единицы должны вводиться после знака умножения, например, **7*kg** (7 килограммов), **2*hr** (2 часа), **4*joule** (4 джоуля). В любой системе единиц каждой первичной единице соответствует одна производная единица. Например, первичной единице L в системе MKS соответствует производная единица m (метр), в американской системе - ft (фут), а в системе CGS – ст (сантиметр). Если в рабочем документе присутствует определение вида a:=1L, то в системе MKS переменная **a** будет равняться 1 метру, в американской системе – 1 футу, а в системе CGS – 1 сантиметру. Пересчет численного значения переменной а производиться не будет. Если же переменная b определена как b:=1*km, то при изменении системы единиц будет выполнен пересчет численного значения. Таким американской образом, в MKS $1 \times km = 1000 \times m$ В системе 1*km=3280.8399*ft, а в CGS 1*km=100000*cm. Производные единицы вводятся в документ MathCAD с помощью диалогового окна Insert Unit командой Units (Единицы) опции Math (Математика) главного меню (см. рис. 1.4) или набором их имен на клавиатуре.



Рис. 1.4. Диалог выбора размерных величин

Имена переменных и функций в MathCAD могут содержать: прописные и строчные латинские, греческие и русские буквы; арабские цифры; символы _, N_2 , % и ∞ , апостроф (на одной клавише со знаком ~). На имена переменных и функций накладываются следующие ограничения: они не могут включать пробелы; не могут начинаться с цифры и русских букв \mathbf{r} и \mathbf{H} ; все символы в имени должны быть напечатаны шрифтом одной гарнитуры, размера и начертания. Если имя переменной или функции содержит недопустимые символы, то будет выведено сообщение об ошибке Not a name (Не является именем).

MathCAD не делает различий между именами переменных и функций. Таким образом, если определить вначале f(x), а затем переменную f, то использование f(x) в дальнейшем будет невозможным. Некоторые имена MathCAD уже использует для встроенных констант, единиц измерения и функций. Эти имена можно переопределить, но при этом уничтожатся их встроенные значения. Одно и то же имя, написанное символами верхних и нижних регистров или отличными шрифтами, воспринимается как различные имена. Если поместить точку в имени переменной, то MathCAD все следующее за ней отобразит как буквенный нижний индекс.

Одиночное число в MathCAD называется скаляром. Столбец или строка чисел называется вектором, а таблица чисел – матрицей. Общий термин для вектора или матрицы – массив. Определение переменной как массива аналогично определению скаляра. Элементом массива может быть как скаляр, так и другой массив. Размер массива задается с помощью команды Matrices...(Матрицы...[Ctrl]+М) опции Math (Математика) главного меню или указанием на первую клетку панели матричных операторов. При этом появляется диалоговое окно, где указываются число строк и столбцов в Можно обращаться к отдельному элементу массива, используя массиве. нижние индексы, которые задаются с помощью наборной панели матричных операторов или нажатием клавиши [. Кроме того, используя верхний индекс, можно обращаться к отдельному столбцу массива. Не следует путать нижние индексы массивов с буквами нижних индексов. Хотя они выглядят одинаково, но буквы нижних индексов являются частью имени. Размер массива зависит от объема доступной памяти и никогда не может превышать восьми миллионов элементов; массивы и векторы, имеющие более чем девять строк и столбцов, отображаются в виде таблицы с полосами прокрутки.

Дискретный аргумент – это переменная, которая принимает ряд значений при каждом ее использовании. Чтобы определить дискретный переменной, аргумент, необходимо напечатать ИМЯ сопровождаемое многоточием и диапазоном значений. Например, j:= 0..15 указывает, что j принимает значения 0,1,2,...,15. По умолчанию приращение принимается равным 1. MathCAD допускает дискретные аргументы со значениями, расположенными в любом диапазоне и меняющимися с произвольным приращением, например, **k** := 1,1.1..2 или **k**:=**a**,**a**-0.1..**b**. Дискретный аргумент задается с помощью наборной панели матричных операторов (клавиша **m..n**) или нажатием клавиши ; . Следует обратить внимание на то, что если для дискретного аргумента используется дробное приращение, нельзя использовать дискретный аргумент как нижний индекс, т.к. нижние индексы должны быть целыми числами. Если дискретный аргумент используется в выражении, MathCAD вычисляет выражение для каждого значения дискретного аргумента.

1.6. Операторы, функции и выражения

Обыкновенная переменная отличается от встроенной переменной тем, что она должна быть предварительно определена, т.е. ей необходимо присвоить значение. В качестве оператора присваивания используется знак := , который вводится нажатием клавиш [Shift]+[:] или с помощью наборной панели операторов отношения. Неопределенная переменная отмечается черным фоном и ее нельзя использовать. Однако можно обеспечить глобальное присваивание, которое производится в любом месте документа. Оно осуществляется с помощью знака ≡ на наборной панели и действует до последующего локального присваивания. Переменные могут быть размерными, и для присваивания значений таким переменным используется знак := .

К скалярным операторам также относятся: вывод значений ($\mathbf{x} =$), сложение ($\mathbf{x} + \mathbf{y}$), вычитание ($\mathbf{x} - \mathbf{y}$), деление ($\frac{\mathbf{x}}{\mathbf{y}}$), умножение ($\mathbf{x} \cdot \mathbf{y}$), возведение в степень ($\mathbf{x}^{\mathbf{y}}$), вычисление квадратного корня ($\sqrt{\mathbf{x}}$), вычисление корня степени п ($^{\mathbf{n}}\sqrt{\mathbf{x}}$), вычисление факториала (\mathbf{n} !), вычисление модуля числа ($|\mathbf{x}|$), вычисление комплексно сопряженного числа (\mathbf{z}). Данные операции могут вводиться как с клавиатуры, так и с помощью наборной панели арифметических операторов.

Система MathCAD содержит ряд расширений скалярных операторов, шаблоны которых изображены на наборной панели объектов высшей математики.

Операторы, предназначенные для сравнения двух величин, называются операторами отношения, или логическими операторами. Эти операторы имеют в основном общепринятые обозначения: >, ≥, <, ≤, ≠, = ([Ctrl]+[=]). Все операторы отношения могут вводиться как самостоятельно с клавиатуры в место расположения курсора, так и с помощью наборной панели, но в этом случае по обе стороны от них появляются маленькие темные прямоугольники, которые являются полями для ввода подлежащих сравнению выражений. Выражения с логическими операторами возвращают логическое значение, которое равно единице, если условие выполнено, и нуль, если оно не выполнено. Можно объединять логические операторы, при этом скалярные операторы умножения (·), действует подобно логическому \mathbf{u} , а сложения (+) – логическому \mathbf{u} . Например, (x<1)·(x>0), (x<1)+(x>0).

Система MathCAD содержит расширенный набор встроенных элементарных функций. Функции задаются своим именем и значением аргумента в круглых скобках. Аргументы и значения функций могут быть действительными или комплексными числами, все углы измеряются в радианах, а многозначные функции и функции с комплексными аргументами всегда возвращают главное значение.

Тригонометрические функции

sin(z) – синус.

 $\cos(z)$ – косинус.

tan(z) – тангенс.

sec(z)	– секанс.
$\csc(z)$	- косеканс.
cot(z)	– котангенс.
Гиперболи	ические функции
sinh(z)	– гиперболический синус.
cosh(z)	– гиперболический косинус.
tanh(z)	– гиперболический тангенс.
sech(z)	– гиперболический секанс.
csch(z)	– гиперболический косеканс.
coth(z)	– гиперболический котангенс.
Обратные	тригонометрические функции
asin(z)	 обратный тригонометрический синус.
acos(z)	 обратный тригонометрический косинус.
atan(z)	– обратный тригонометрический тангенс.
Обратные	гиперболические функции
asinh(z)	 обратный гиперболический синус.
acosh(z)	 обратный гиперболический косинус.
atanh(z)	 обратный гиперболический тангенс.
Показател	ьная и логарифмические функции
exp(z)	– экспоненциальная функция.
ln(z)	– натуральный логарифм (по основанию е).
log(z)	– десятичный логарифм (по основанию 10).
Функции	усечения и округления
Re(z)	- выделение действительной части z.
Im(z)	– выделение мнимой части z.
arg(z)	 вычисление угла в комплексной плоскости между положительным направлением вещественной оси и числом z.
floor(x)	 – наибольшее целое число ≤х, где х–вещественное.
ceil(x)	– наименьшее целое число ≥х, где х–вещественное.
mod(x,y)	 остаток от деления х на у. Знак результата совпадает со знаком числа х.

angle(x,y) – угол (в радианах) между осью абсцисс и вектором с координатами (x,y).

Наряду с элементарными функциями в системе MathCAD содержатся встроенные *специальные* математические функции. Наличие таких функций значительно расширяет возможности системы MathCAD.

Функции Бесселя

J0(x)	– возвращает J ₀ (x), где x - вещественный.	
J1(x)	– возвращает J ₁ (x), где х – вещественный.	
Jn(m,x)	– возвращает Ј _m (х), где х – вещественный, 0≤m≤100.	
Y0(x)	– возвращает Y ₀ (x), где х – вещественный, х>0.	
Y1(x)	– возвращает Y ₁ (x), где х –вещественный, х>0.	
Yn(m,x)	 возвращает Y_m(x), где х –вещественный, х>0, 0≤m≤100. 	
I0(x)	– возвращает I ₀ (x), где x - вещественный.	
I1(x)	– возвращает I ₁ (x), где х - вещественный.	
In(m,x)	- возвращает І _m (х), где х - вещественный, 0≤m≤100.	
K0(x)	– возвращает К ₀ (х), где х – вещественный, х>0.	
K1(x)	– возвращает K ₁ (x), где х – вещественный, х>0.	
Kn(m,x)	- возвращает К _m (х), где х -вещественный, х>0, 0≤m≤100.	
Кусочно-н	епрерывные функции	
if(x,y,z)	 возвращает значение у, если х≠0 (истина), и значение z, если х=0 (ложь). 	
until(x,z)	- возвращает значение z, пока выражение x не станет отрицательным (x должно содержать дискретный аргумент).	
$\Phi(\mathbf{x})$	- ступенчатая функция Хэвисайда: возвращает 1, если х≥0, и возвращает 0, если х<0.	
δ(m,n)	 – символ Кронекера: возвращает 1, если m=n, и возвращает 0, если m≠n (т и n должны быть целочисленными). 	
ε(i,j,k)	 полностью антисимметричный тензор ранга 3: возвращает 0, если любые два аргумента одинаковы, 1 для четных перестановок, –1 для нечетных перестановок (i, j, k должны быть целыми числами между ORIGIN и ORIGIN+2 включительно). 	

Специальные функции

- erf(x) возвращает значение интеграла ошибок в x, где x должен быть вещественным.
- Г(z) возвращает значение эйлеровой гамма-функции в z.

Функции дискретных преобразований

- fft(v) возвращает вектор, размером 1+2ⁿ⁻¹, с комплексными элементами, который является дискретным преобразованием Фурье вещественного вектора v, размером 2ⁿ.
- ifft(v) возвращает вещественный вектор, размером 2ⁿ, который является обратным преобразованием Фурье вектора v, размером 1+2ⁿ⁻¹, с комплексными элементами.
- cfft(A) возвращает дискретное преобразование Фурье массива с комплексными элементами.
- icfft(A) возвращает обратное дискретное преобразование Фурье массива с комплексными элементами.
- wave(v) возвращает дискретное волновое преобразование вещественного вектора v, возвращаемый вектор и аргумент имеют размер 2ⁿ.
- iwave(v) возвращает обратное дискретное волновое преобразование вещественного вектора v, возвращаемый вектор и аргумент имеют размер 2ⁿ.

Кроме того, существуют функции FFT, IFFT, CFFT, ICFFT, peanusyющие альтернативные формы дискретного преобразования Фурье. Они используются аналогично функциям fft, ifft, cfft, icfft.

Несмотря на довольно широкий набор встроенных функций, всегда возникает необходимость расширить систему новыми функциями, представляющими интерес для пользователя. *Функции пользователя* вводятся с помощью следующего выражения:

Имя_функции (Список_параметров) := Выражение

Имя функции задается как любой идентификатор. В скобках указывается список параметров функции – это перечень используемых в выражении переменных, разделенных запятыми. Выражение – это любой набор доступных системе операторов и функций с операндами и аргументами, указанными в списке параметров. Следует отметить особый статус переменных, указанных в списке параметров функций пользователя. Эти переменные являются локальными, поэтому они могут не определяться до задания функций.

Фактически их указание в списке параметров является заданием этих переменных. Локальные переменные могут использоваться только в выражении, описывающем функцию. Их имена могут совпадать с именами глобальных переменных, введенных ранее. Но при этом по выходе из блока задания функции значения этих переменных будут сохранять ранее заданные (для глобальных переменных) значения.

1.7. Редактирование выражений

Редактирование выражений можно осуществлять как с применением мыши, так и клавиатуры. В любом случае, прежде всего, полезно знать функции графического курсора – маркера. Он при перемещении по документу может принимать одну из трех приведенных ниже форм:

| – курсор в виде вертикальной черты (маркер ввода) служит для указания на отдельные элементы *блоков* и обычно используется для ввода данных и заполнения шаблонов;

+ – крестообразный курсор (визир) служит для указания места для новых блоков;

□ – курсор в виде синей рамки предназначен для выделения отдельных частей выражения или выражения целиком. Рамка имеет срезанный верхний уголок, указывающий на направление последующего ввода (слева направо). Нажатие клавиши **Ins** меняет цвет рамки на красный и перемещает уголок в левый верхний угол, что означает возможность набора слева от выделенного рамкой выражения.

Клавиши для выделения:

[1]	 превращает маркер в выделяющую рамку и расширяет ее;
[↓]	 сужает выделяющую рамку, если же она содержит имя или число, то превращает рамку в маркер;
$[\rightarrow]$	 перемещает маркер или рамку вправо;
[←]	 перемещает маркер или рамку влево;
[Shift]+[↑]	 выводит курсор из выражения в свободное поле, делая его крестообразным;
$[Shift]+[\downarrow]$	– выводит курсор из выражения вниз в свободное поле;
$[Shift]+[\rightarrow]$	– выводит курсор из выражения вправо в свободное поле;
[Shift]+[←]	– выводит курсор из выражения влево в свободное поле;
[Space]	 заключает в рамку операнд, действует как несколько нажатий клавиши [↑] и выводит курсор из выделенного выражения;

[Ins]	_	перемещает срез рамки из правого верхнего угла в левый верхний угол.
Клавиши и их	кк	омбинации для управления редактированием:
[Tab]	_	в тексте перемещает курсор на начало следующего слова, в уравнении управляет выделением частей блока (в частности, выделяя выражения в скобках);
[Shift]+[Tab]	_	в тексте перемещает курсор в начало очередного слова, в уравнении управляет выделением частей блока;
[PgUp]	_	перемещает курсор и вызывает скроллинг на пять строк вверх;
[PgDn]	_	перемещает курсор и вызывает скроллинг на пять строк вниз;
[Ctrl]+[PgUp]	_	вызывает скроллинг на одно окно вверх;
[Ctrl]+[PgDn]	_	вызывает скроллинг на одно окно вниз;
[Home]	—	устанавливает курсор в начало предшествующего блока;
[Ctrl]+[Home]	_	вызывает скроллинг с установкой курсора в начало документа;
[Ctrl]+[End]	_	вызывает скроллинг с установкой курсора в конец документа.
Клавиши редактирования:		
[Shift]+[F5]	_	замена одной подстроки на другую;
[Ctrl]+[F9]	—	вставка пустой строки;
[Ctrl]+[F10]	_	удаление пустой строки;
[Alt]+[Bksp]	_	отмена последней операции редактирования;
[Ctrl]+[Ins]	_	копирование (Сору) выделенного объекта в буфер обмена;
[Shift]+[Del]	_	перенос (Cut) выделенного объекта в буфер обмена;
[Shift]+[Ins]	_	копирование объекта из буфера обмена в окно;
[Ins]	_	включение режима вставки.

Математические выражения не столько набираются, сколько конструируются. При этом учитываются определенная структура выражений и иерархия выполняемых операций. Конструирование выражений облегчается средствами выделения отдельных фрагментов выражений. Полезно помнить, что все, что попадает в ходе выделения в рамку, оказывается операндом для следующего вводимого оператора. Операторы возведения в степень, извлечения корня и деления являются *цепкими* операторами. После их ввода все, что набирается следом, становится показателем степени, подкоренным выражением или знаменателем. Для выхода из этой ситуации необходимо дважды нажать клавишу [Space] (пробел).

2. Работа с файлами и документами

2.1. Подменю <u>F</u>ile главного меню

Опция <u>File</u> (Файл) главного меню служит для работы с файлами документов. Файлом называют имеющую ИМЯ упорядоченную последовательность данных или кодов программ, размещенную на том или ином носителе информации. Файлы документов в MathCAD имеют расширение .mcd, которое указывается сразу после имени файла. Такие файлы имеют текстовый формат и их легко прочитать и модифицировать любым текстовым редактором. Файлы документов содержат полный текст программы, которая координаты блоков, фактическое ИХ содержание, характер описывает выполняемых операций, форматы предоставления информации и т.д. Таким образом, файл является, по сути, программой, записанной на внутреннем языке программирования системы. Файлы могут содержать и результаты вычислений.

Предусмотрена возможность записи документов и в особом формате RTF, созданном для хранения сложных многокомпонентных данных (содержащих тексты и графику). Важно отметить, что даже при записи документов со сложными рисунками используется не запись их BitMaP-копии, а именно программы вывода документа. Поэтому файлы с расширением .mcd невелики по размеру, и их легко передавать по современным средствам телекоммуникаций, например, по сети Internet.

Помимо обычных операций по работе с файлами (их запись на диск и загрузка с диска) предусмотрены возможности распечатки документов принтерами различного типа.

Опция <u>File (Файл)</u> главного меню содержит следующие команды:

<u>New (Создать [F7])</u> – открыть окно для нового документа;

<u>Ореп...</u> (Открыть... [F5]) – открыть ранее записанный документ;

<u>Save (Coxpaнить [F6])</u> – сохранить на диске текущий документ;

Save <u>A</u>s... (Сохранить как...) – сохранить на диске текущий документ под новым именем или путем доступа;

Export <u>Worksheet...</u> (Экспорт...) – сохранить рабочий документ в формате RTF;

Insert... (Вставить документ...) – вставить в документ новый фрагмент с диска; **Close (Закрыть [Ctrl]+[F4])** – закрыть документ;

Open URL... (Открыть URL...) – открыть документ MathCAD, доступный через Internet;

Get F<u>r</u>om Notes... (Получить из Notes...) – открыть документ MathCAD, хранящийся в базе данных Lotus Notes;

Save <u>To</u> Notes... (Сохранить в Notes...) – сохранить документ MathCAD в существующей базе данных Lotus Notes;

<u>Mail...</u> (Послать почту...) – отправить текущий документ по электронной почте;

Save Configuration... (Сохранить конфигурацию...) – записать текущую конфигурацию среды MathCAD в файл конфигурации;

Execute Configuration File... (Применить конфигурацию...) – выполнить файл конфигурации;

Associa<u>t</u>e Filename... (Присоединить к файлу...) – связать выделенную переменную с файлом данных;

Page Set<u>up</u>... (Параметры страницы...) – установить левый и правый отступы на странице;

Print Preview... (Просмотр...) – просмотреть документ перед печатью;

<u>Print... (Печать... [Ctrl]+O)</u> – печать документа;

<u>E</u>xit (Выход [Alt] +[F4]) – выйти из среды MathCAD.

2.2. Подменю <u>E</u>dit главного меню

Выбор опции <u>Edit (Редактирование)</u> в главном меню приводит к выводу обширного ниспадающего подменю, содержащего следующие команды:

<u>Undo Last Edit (Отменить изменения [Alt]+[Bksp])</u> – отменить последнюю операцию редактирования;

Си<u>т</u> (Вырезать [Ctrl]+Х) – переместить выделенную область в буфер обмена;

<u>Сору</u> (Копировать [Ctrl]+C) – копировать выделенную область в буфер обмена;

Clear (Очистить) – удалить выделенную область;

<u>Paste</u> (Вставить [Ctrl]+V) – вставить содержимое буфера обмена в документ;

Paste Special... (Специальная вставка...) – вставить содержимое буфера обмена в различном формате (в формате MathCAD или BitMap);

<u>Regions (Области)</u> – вывод подменю:

View Regions (Показать области) – показать области путем их подсветки;

Select <u>All Regions</u> (Выделить все области) – выделить все области в документе;

<u>Separate Regions (Разделить области [Ctrl]+S)</u> – разделить перекрывающиеся области;

Align Regions (Выровнять области) – вывод подменю:

Horizontal (Горизонтально) – выровнять выделенные области вдоль горизонтальной линии, расположенной посередине между верхними краями высшей и низшей из выделенных областей;

<u>Vertical</u> (Вертикально) – выровнять выделенные области вдоль вертикальной линии, расположенной посередине между левыми краями самой правой и самой левой из выделенных областей;

Include... (Внедрить...) – вставить определения переменных и функций из другого рабочего документа в текущий документ;

Link (Связать) – вывод подменю:

<u>N</u>ew (Новая) – создать гипертекстовую ссылку на другой рабочий документ;

<u>Erase (Удалить)</u> – удалить все гипертекстовые ссылки, связанные с текущим выделением;

Lock Regions... (Закрыть область...) – вывод подменю:

Set Lock Area (Область) — определить защищаемую область рабочего документа;

Lock Area... (Закрыть...) – включить защиту области;

Unlock Area... (Открыть...) – редактировать защищаемую область;

Ins/<u>D</u>el Blank Lines... (Вставить/Удалить чистые строки...) – вставить/удалить строки;

Insert Pagebreak (Вставить разрыв страницы) – вставить независимый от форматирования разрыв страниц;

Right <u>Margin (Правое поле)</u> – вывод подменю:

Set (Установить) – установить правый отступ;

Clear (Удалить) – убрать правый отступ;

<u>H</u>eaders/Footers... (Колонтитулы...) – определить верхние/нижние колонтитулы в документе;

<u>F</u>ind... (Найти... [Ctrl]+[F5]) – найти заданную текстовую или математическую строку;

<u>**Replace... (Заменить... [Shift]+[F5])** – найти и заменить математическую или текстовую строку;</u>

<u>Go to Page... (Перейти к странице...)</u> – расположить начало указанной страницы в начале рабочего документа MathCAD.

2.3. Подменю <u>Т</u>ехt главного меню

Система MathCAD снабжена достаточно простым текстовым редактором. Текстовые комментарии являются блоками и занимают определенную область в окне. С позиции работы MathCAD текстовые блоки являются неисполняемыми объектами.

Для работы с текстовыми комментариями, в опции <u>**T**ext</u> (**Tекст**) главного меню представлены следующие команды:

Create Text <u>Region</u> (Создать текстовую область ["]) – создать текстовую область с началом в месте расположения курсора;

Create Text <u>Paragraph</u> (Создать текстовый параграф [Ctrl]+T) – создать текстовый параграф с первой строкой в месте расположения курсора;

Embed <u>Math</u> (Внедрить формулы) – создать математическую область внутри текстовой области или параграфа;

Change <u>F</u>ond... (Изменить шрифт...) – изменить шрифт для выделенного комментария;

Change Paragraph Format... (Изменить формат параграфа...) – изменить способ выравнивания и левый отступ выделенного параграфа;

Change Default (Изменить по умолчанию) – вывод подменю:

<u>Font... (Шрифт)</u> –изменить шрифт всего текста, кроме тех фрагментов, для которых текст изменен командой Change Fond... (Изменить шрифт...);

Paragraph Format...(Формат параграфа...) – изменить принятые по умолчанию способ выравнивания и левый отступ выделенных параграфов;

Check <u>Spelling...(Орфография...)</u> – проверить орфографию (только для англоязычных текстов).

Текстовая область – это область текста изменяемого размера, ограниченного длиной наибольшей строки. Она определяется нажатием клавиши [Enter]. Текстовый параграф по размеру равен ширине страницы.

Команда Change <u>F</u>ond... (Изменить шрифт...) доступна только по отношению к выделенному фрагменту текста. При использовании этой команды появляется диалоговое окно <u>Fond</u> (Шрифт), которое содержит три переключателя для выбора:

Font (Шрифт) –набор шрифтов;

Font Style (Стиль) – стиль шрифтов:

Bold (Полужирный) – полужирный шрифт;

Italic (Наклонный) – курсив;

Underline (Подчеркивание) – шрифт с подчеркиванием снизу.

По умолчанию стиль шрифтов нормальный.

Команда Change Paragraph Format... (Изменить формат параграфа...) вызывает диалоговое окно, которое имеет опции: Indent (Отступы) – задание абзацного отступа:

<u>All Lines (Для всех строк)</u> – установка отступа для всех строк;

<u>F</u>irst Lines (Для первых строк) – установка отступа для первых строк;

Alignment (Выравнивание) –задание типа равнения строк:

Left (Слева) – по левому краю абзаца;

<u>Right (Справа)</u> –по правому краю абзаца;

<u>C</u>enter (По центру) – по центру.

2.4. Подменю Window главного меню

Система MathCAD позволяет одновременно работать с восемью задачами, под каждую из которых она отводит отдельное окно. То окно, с которым происходит работа в настоящий момент, является текущим и активным.

В опции <u>Window (Окно)</u> главного меню сосредоточены следующие команды для работы с окнами:

<u>Cascade</u> (Каскад) – расположить окна документов друг под другом таким образом, чтобы видны были заголовки;

Tile <u>H</u>orizonntal (По горизонтали) – расположить окна документов горизонтально так, чтобы они не перекрывались;

<u>Tile Vertical (По вертикали)</u> – расположить окна документов по вертикали так, чтобы они не перекрывались;

<u>Arrange Icons (Упорядочить значки)</u> – упорядочить размещение пиктограмм документов вдоль нижней границы окна приложения;

Zoom... (Масштаб...) – изменить размеры изображения документа;

<u>Refresh (Обновить [Ctrl]+R)</u> – восстановить документ;

Animation (Анимация) – создание и проигрывание анимационных файлов:

<u>Create...(Создать)</u> – создать анимационный клип;

<u>Playback... (Воспроизвести)</u> – воспроизвести анимационный клип;

Hide <u>P</u>alette (Убрать панель символов) – убрать панель символов (когда панель убрана, эта опция отмечается);

Hide Tool Bar (Убрать панель инструментов) – убрать верхнюю панель инструментов;

Hide Font Bar (Убрать панель шрифтов) – убрать панель шрифтов;

Change Colors...(Изменить цвета...) – изменить цвета:

Background Color...(Фон...) – изменить цвет фона окна приложения;

<u>Text Color...(Текст...)</u> – изменить цвет шрифта в текстовых областях;

Equation Color...(Формулы...) – изменить цвет формул и чертежей;

Equation <u>Highlight</u> Color...(Цвет подсвеченного выражения...) – изменить цвет подсветки выражений;

<u>Annotation Color...(Аннотации...)</u> – изменить цвет всех аннотаций к электронным книгам.

Любая из команд опции Change Colors...(Изменить цвета...) выводит диалоговое окно, которое открывает доступ к командам цветового оформления документов.

2.5. Открытие рабочего документа

Чтобы открыть ранее сохраненный рабочий документ, нужно выбрать команду **Ореп...** (Открыть... [F5]) из опции <u>File</u> (Файл). Система MathCAD запросит имя файла. Текущая директория указана под словом Папки. Если файл, который нужно открыть, находится в этой директории, то следующий шаг можно пропустить. В противном случае нужно переместить указатель в список просмотра папок и дважды щёлкнуть на папке, содержащей нужный файл. Этот список включает также все папки, находящиеся в текущей директории. Чтобы переключиться на другой диск, нужно нажать кнопку со стрелкой рядом с окошком Диски и выбрать нужный диск. Рядом со списком директорий находится список просмотра имен всех файлов, находящихся в текущей директории и имеющих расширение .mcd. Двойной щелчок по имени файла открывает его. Можно также выбрать имя файла, а затем нажать кнопку

Open		[<u>?</u>]X
<u>И</u> мя файла:]*.mcd	Папки: e:\mathcad mathcad mATHCAD CLIPS EFI HANDBOOK MAPLE	ОК Отмена
<u>Т</u> ип Файлов: <u> Mathcad Files(*.mcd)</u>	Диски: е:	

Рис. 2.1. Диалог открытия рабочего документа

ОК. В нижней части подменю <u>File</u> (Файл) MathCAD ведет список файлов, открывавшихся последними. Выбор имени файла из этого списка сразу открывает файл.

2.6. Открытие рабочего документа через Internet

Если рабочий документ MathCAD находится на World Wide Web, то его нужно открывать с использованием команды Open URL... (Открыть URL...) опции <u>File (Файл)</u>. Для этого необходимо набрать URL (Uniform Resource Locator – Единообразный Адрес Ресурса) в окне диалога. Он содержит имя компьютера, хранящего рабочий документ, и путь к нему в файловой системе этого компьютера.

Фирма MathSoft предоставляет сборник связанных друг с другом рабочих документов MathCAD. Чтобы получить доступ к этому сборнику, нужно выбрать пункт QuickSheets (Шпаргалки) из меню <u>Help (Справка)</u> и щёлкнуть по MathSoft Web Site. Чтобы использовать эту возможность, нужно использовать стек TCP/IP, иначе появится сообщение об ошибке "Не удаётся найти WIN-SOCK.DLL". В этом случае необходимо установить стек TCP/IP, но не все реализации стека TCP/IP обладают необходимыми характеристиками. Точно известны как совместимые следующие реализации: Microsoft's Wolverine^{тм}, FTP Softwares ONnet^{тм} и NetManages Chameleon^{тм}.

2.7. Сохранение рабочего документа

В опции <u>File</u> (Файл) главного меню имеются две команды, касающиеся сохранения файлов в локальной файловой системе: <u>Save</u> (Coxpaнить [F6]) и Save <u>As...</u> (Coxpaнить как...). Для сохранения рабочего документа, ранее никогда не сохранявшегося, можно выбрать любую из них, в результате чего появится диалоговое окно Save <u>As...</u> (Coxpaнить как...) (см. рис. 2.2), куда нужно ввести имя файла. Чтобы зафиксировать изменения в рабочем

Save As			[?] IX
Папка:	🔁 Mathcad		
Clips Efi Handbook Maple Qsheet Spell	C Userefi		
<u>И</u> мя файла:	<u> </u>		Со <u>х</u> ранить
<u>Т</u> ип файла:	Mathcad Files (*.mcd)	v	Отмена

Рис. 2.2. Сохранение рабочего документа

документе, не уничтожая его исходного варианта, нужно также выбрать команду Save <u>A</u>s... (Сохранить как...) из опции <u>File (Файл)</u>.

Чтобы задать новое имя файла, необходимо:

- указать в окне Папка путь, показывающий, где нужно сохранить файл;
- набрать имя файла в окне ввода (по умолчанию MathCAD добавляет к имени файла расширение .mcd);
- нажать кнопку ОК.

Для перезаписи исходного рабочего документа его измененной версией нужно выбрать команду <u>Save (Coxpaнutb [F6])</u>, в результате чего MathCAD уничтожает исходную копию рабочего документа и записывает новую копию, показываемую в рабочем окне.

2.8. Пересылка рабочих документов по электронной почте

Если используемый компьютер подключен к системе электронной почты, совместимой либо с сс:Mail, либо с Microsoft Mail, то эту систему можно использовать для пересылки рабочих документов вместе с коротким посланием. При пересылке почтового сообщения получатель получает рабочий документ в виде файла-приложения.

Чтобы переслать рабочий документ по электронной почте, нужно:

- открыть рабочий документ, который нужно переслать;
- выбрать команду <u>Mail... (Послать почту...)</u> из опции <u>File (Файл).</u>

После этого появляется последовательность диалоговых окон, принадлежащих используемой системе электронной почты, запрашивающих пароль, адрес и все прочее, необходимое для пересылки послания и зависящих от конкретной системы. Если система электронной почты не отвечает, то, может быть, придется добавить путь в каталог, содержащий почтовый сервер, к общему пути. При наличии двух серверов, использующих одинаковые протоколы, например SMI, MathCAD использует сервер, записанный первым в пути.

2.9. Экспорт рабочего документа

Рабочий документ MathCAD может быть экспортирован в формат RTF, распознаваемый большей частью современных текстовых процессоров. Для этого необходимо:

- перейти в самый конец видимого в окне документа, чтобы обновить все результаты вычислений;
- выбрать команду Export Worksheet... (Экспорт...) из опции File (Файл);
- указать в окне Папка соответствующий путь, куда нужно экспортировать RTF-файл;

- набрать имя файла в окне ввода (по умолчанию MathCAD добавляет к имени файла расширение .rtf);
- выбрать команду Export Worksheet... (Экспорт...) из опции File (Файл);
- нажать кнопку ОК.

После открытия такого RTF-файла в текстовом процессоре все области MathCAD будут расположены одна над другой и выровнены по левому краю документа. После этого можно перемещать эти области текстовым процессором куда угодно. После того, как области MathCAD загружены в текстовый процессор, редактировать уравнения и чертежи можно только как графические изображения, в то время как текст можно редактировать обычным образом.

2.10. Импорт и экспорт текста

Текст MathCAD форматируется с использованием спецификации RTF. Это означает возможность экспортировать текст из текстовых областей MathCAD или параграфов в любой текстовый процессор или другую программу, воспринимающую файлы в формате RTF. Во многие текстовые редакторы, запущенные под Windows одновременно с MathCAD, можно переносить текст непосредственно через буфер обмена. Для этого необходимо:

- щёлкнуть в текстовой области или параграфе, чтобы поместить туда маркер ввода;
- перемещаясь, выделить текст, который нужно экспортировать;
- выбрать команду <u>Сору</u> (Копировать [Ctrl]+C) из опции <u>E</u>dit (Редактирование);
- щёлкнуть в окне программы, куда требуется перенести текст, и вставить его из буфера.

Можно также импортировать текст из большинства других прикладных программ под Windows. Чтобы это сделать необходимо:

- поместить текст в буфер;
- щёлкнуть в свободном месте рабочего документа MathCAD, курсор при этом должен иметь вид крестика;
- в MathCAD выбрать команду <u>Paste</u> (Вставить [Ctrl]+V) из опции <u>Edit</u> (Редактирование).

Система MathCAD создаст параграф, содержащий текст из буфера обмена. Если текст содержит RTF коды форматирования, MathCAD отформатирует текст в соответствии с ними.

Windows-приложения, работающие как OLE серверы, позволяют внедрить текст в рабочий документ. Это означает, что при двойном щелчке мышью на внедренном тексте запускается текстовый процессор, который создал текст, и копия текста помещается в окно документа. Последующее редактирование можно делать в текстовом процессоре. Чтобы внедрить текст необходимо:

- скопировать текст, который нужно вставить из текстового процессора, в буфер обмена;
- щёлкнуть в рабочем документе MathCAD и выбрать команду Paste Special... (Специальная вставка...) из опции <u>Edit</u> (Редактирование) MathCAD.

Чтобы открыть текстовый процессор, связанный с внедрённым текстом, нужно дважды щёлкнуть на этом тексте в рабочем документе MathCAD. Можно также экспортировать весь рабочий документ, включая уравнения и графики, в RTF формат.

2.11. Поиск текста в рабочем документе

Команда <u>Find...</u> (Найти... [Ctrl]+[F5]) работает как в текстовых, так и в математических областях. Когда требуется осуществить поиск строки символов, MathCAD ищет эту строку как часть имени переменной или функции либо как фрагмент текста в текстовой области или параграфе.

Для поиска строки символов необходимо:

- выбрать команду <u>Find...</u> (Найти... [Ctrl]+[F5]) из опции <u>E</u>dit (Редактирование), при этом появится диалоговое окно;
- ввести строку для поиска в поле <u>Find (Найти);</u>
- щёлкнуть на Next (Вперед) или Previous (Назад), чтобы задать поиск строки непосредственно после или перед текущим расположением маркера.

Когда MathCAD находит искомую строку в текстовой области или параграфе, она показывается в негативном изображении; если строка обнаруживается в математической области, на ней устанавливается курсор. Диалоговое окно остается на экране, чтобы можно было продолжить поиск. Команда <u>Find...</u> (Найти...) является чувствительной к регистру, но не чувствительной к шрифту.

2.12. Вставка одного рабочего документа внутрь другого

Когда нужно использовать формулы и вычисления из одного рабочего документа в другом, можно, конечно, открыть оба листа и воспользоваться командами <u>Copy</u> (Копировать [Ctrl]+C) и <u>Paste</u> (Вставить [Ctrl]+V) из опции <u>Edit</u> (Редактирование), однако для переноса многих областей этот способ достаточно обременителен. Есть два других способа предоставить формулы из одного документа для использования в другом. Выбор подходящего зависит от

того, нужно ли, чтобы перенесенные области действительно были видны во втором рабочем документе или чтобы второй рабочий документ только вел себя так, как будто эти области в нем присутствуют.

Первый способ называется вставка рабочего документа, а второй внедрение рабочего документа. Вне зависимости от того, какой способ выбран, первый шаг один и тот же: щёлкнуть мышью на свободном месте там, где нужно вставить рабочий документ. После этого курсор должен принять вид крестика.

Чтобы вставить один рабочий документ в другой, нужно:

- выбрать команду <u>Insert...</u> (Вставить документ...) из опции <u>File</u> (Файл), после чего появится такое же диалоговое окно, как при открытии любого рабочего документа;
- задать в диалоговом окне путь и имя вставляемого рабочего документа;
- нажать кнопку ОК.

После нажатия кнопки OK MathCAD сдвигает все области в окне рабочего документа ниже места расположения курсора-крестика, чтобы освободить место, после чего переносит туда содержимое вставляемого документа.

Если рабочий документ нужно *внедрить*, то необходимо, открыв оба листа одновременно, сделать следующее:

- открыть рабочий документ, который нужно внедрить, выбрав команду <u>Ореп...</u> (Открыть... [F5]) из опции <u>File</u> (Файл), щелкнуть по окну, в которое нужно внедрить рабочий документ, и установить курсор в то место, куда нужно внедрить документ;
- выбрать команду <u>Include...</u> (Внедрить...) из опции <u>Edit</u> (Редактирование), после чего появится диалоговое окно Include;
- в списке открытых документов выбрать тот, который нужно внедрить (в списке отсутствуют все открытые документы без имени, поэтому документ должен быть сохранен хотя бы один раз перед тем, как его можно будет вставить);
- нажать кнопку ОК.

После нажатия кнопки ОК MathCAD вставляет пиктограмму в место расположения указателя-крестика. Все определения во внедрённом рабочем документе будут действительны ниже и правее этой пиктограммы. После двойного щелчка по этой пиктограмме MathCAD показывает внедрённый файл. Эту иконку можно передвигать или исключать, как и любую область MathCAD.

3. Операторы и функции для работы с векторами и матрицами

Для работы с векторами и матрицами система MathCAD содержит ряд операторов и функций. Ниже представлены операторы и функции для работы с векторами и матрицами, причем используются следующие общие обозначения: V – вектор, M,A – матрицы, Z – скаляр. Заметим, что если аргументом является вектор, то это обязательно вектор-столбец.

3.1. Операторы

V1+V2	 сложение двух векторов V1 и V2.
VI-V2	– вычитание двух векторов V1 и V2.
V1*V2	– умножение двух векторов V1 и V2.
-V	 смена знака у элементов вектора V.
-M	 смена знака у элементов матрицы М.
V-Z	 вычитание из вектора V скалярной величины Z.
Z*V, V*Z	 умножение вектора V на скалярную величину Z.
Z*M, M*Z	 – умножение матрицы М скалярную величину Z.
M*V	– умножение матрицы М на вектор V.
M1*M2	– перемножение двух матриц M1 и M2.
$\frac{\mathbf{V}}{\mathbf{Z}}$	 деление вектора V на скалярную величину Z.
$\frac{M}{Z}$	 деление матрицы М на скалярную величину Z.
M ⁿ	– возведение матрицы M в степень n.
$ \mathbf{V} $	– длина вектора V.
$ \mathbf{M} $	 вычисление определителя матрицы М.
\mathbf{V}^{T}	 транспонирование вектора V.
\mathbf{M}^{T}	 транспонирование матрицы М.
Vl×V2	– векторное перемножение двух векторов V1 и V2.
$\overline{\mathbf{V}}$	– вычисление комплексно-сопряженного к V вектора.
$\overline{\mathbf{M}}$	 вычисление комплексно-сопряженной к М матрицы.
$\sum V$	 вычисление суммы элементов вектора V.

M ^{<n></n>}	 выделение n-го столбца матрицы M.
V _n	– выделение n -го элемента вектора V.
$\mathbf{M}_{\mathbf{m},\mathbf{n}}$	 выделение элемента матрицы M, стоящего на пересечении m -й строки и n -го столбца.
a b	 векторизация оператора.
$\mathbf{f}(\mathbf{x})$	 векторизация функции.

Оператор векторизации изменяет действие любого оператора или функции следующим образом. Оператор или функция, требующая скалярный аргумент, при получении в качестве аргумента массива (вектора или матрицы), применятся к каждому элементу этого массива поочередно. Примеры использования оператора векторизации приведены на рис. 3.1.

Задание матриц А и В	$A := \begin{pmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{pmatrix}$ $B := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
Простое произведение матриц и векторизованное (перемножение элементов матриц с совпадающими индексами)	$\mathbf{A} \cdot \mathbf{B} = \begin{pmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{pmatrix} \qquad \qquad \overrightarrow{(\mathbf{A} \cdot \mathbf{B})} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 9 \end{pmatrix}$
Попытка вычислить синус от матрицы А	sin(A) = non-scalar value
Вычисление синуса от каждого элемента матрицы А	$\overrightarrow{sin(A)} = \begin{pmatrix} 0.84147 & -0.7568 & 0.65699 \\ 0.9093 & -0.95892 & 0.98936 \\ 0.14112 & -0.27942 & 0.41212 \end{pmatrix}$

Рис. 3.1. Примеры использования оператора векторизации

3.2. Векторные функции

length (V)	 возвращает длину вектора V.
last(V)	 возвращает индекс последнего элемента вектора V.

- sort (V) сортировка элементов вектора V в порядке возрастания их значений.
- reverse(V) перестановка элементов вектора V в обратном порядке.
- **Re(V)** возвращает вектор, состоящий из действительных частей элементов вектора V.
- Im(V) возвращает вектор, состоящий из мнимых частей элементов вектора V.
- max(V) возвращает максимальный элемент в V. Если V содержит комплексные элементы, то возвращает max(Re(V))+i·max(Im(V)).

min(V) – возвращает минимальный элемент в V. Если V содержит комплексные элементы, то возвращает min(Re(V))+i·min(Im(V)).

3.3. Матричные функции

- augment(M1, M2) объединяет в одну матрицы M1 и M2, имеющие одинаковое число строк, записывая матрицу M2 справа от M1.
- stack (M1, M2) объединяет две матрицы Ml и M2, имеющие одинаковое число столбцов, располагая Ml над M2.
- submatrix (A, ir, jr, ic, jc) возвращает подматрицу, состоящую из элементов, находящихся в строках с ir по jr и в столбцах с ic по jc (ir \leq jr, ic \leq jc).
- identity (n) создает единичную матрицу размерности $\mathbf{n} \times \mathbf{n}$.
- diag (V) создает диагональную матрицу, элементы главной диагонали которой равны элементам вектора V.
- **matrix (m,n,f)** создает матрицу размерности $\mathbf{m} \times \mathbf{n}$, элементы которой вычисляются следующим образом: $\mathbf{A}_{i,j} = \mathbf{f}(i, j)$.
- csort(M, n) перестановка строк матрицы М таким образом, чтобы отсортированным оказался **n**-й столбец.
- rsort(M, n) перестановка столбцов матрицы М таким образом, чтобы отсортированной оказалась **n**-я строка.
- **Re(M)** возвращает матрицу, состоящую из действительных частей элементов матрицы **M**.
- Im(M) возвращает матрицу, состоящую из мнимых частей элементов матрицы M.

max(M)	_	возвращает	максимальный	элемент	В	M.	Если	Μ
		содержит	комплексные	элементы,	тс)]	возвращ	ает
		max(Re(M))+i·max(Im(M)).						

min(M) – возвращает минимальный элемент в М. Если М содержит комплексные элементы, то возвращает min(Re(M))+i·min(Im(M)).

3.4. Функции, возвращающие специальные характеристики матриц

cols (M)	 возвращает число столбцов матрицы М. 						
rows (M)	 возвращает число строк матрицы М. 						
rank (M)	 возвращает ранг матрицы М. 						
tr (M)	– возвращает след квадратной матрицы М (сумму диагональных элементов).						
mean (M)	 возвращает среднее значение элементов массива М. 						
median (M)	 возвращает медиану элементов массива М. 						
norml(M)	 возвращает L1 норму матрицы М. 						
norm2(M)	 возвращает L2 норму матрицы М. 						
norme(M)	 возвращает евклидову норму матрицы М. 						
normi(M)	 возвращает равномерную норму матрицы М. 						
condl (M)	 возвращает число обусловленности матрицы М, вычисленное по норме L1. 						
cond2 (M)	– возвращает число обусловленности матрицы М , вычисленное по норме L2.						
conde (M)	 возвращает число обусловленности матрицы М, вычисленное по норме евклидова пространства. 						
condi(M)	 возвращает число обусловленности матрицы М, основанное на равномерной норме. 						
lu (M)	 выполняет треугольное разложение матрицы М: P·M=L·U, где L и U - соответственно нижняя и верхняя треугольные матрицы. Все четыре матрицы квадратные, одного порядка. 						
qr(A)	 выполняет разложение матрицы A вида A = Q·R, где Q - ортогональная матрица, а R верхняя треугольная матрица. 						

- svd(A) выполняет сингулярное разложение матрицы A размером n×m: A=U·S·V, где U и V - ортогональные матрицы размерами m×m и n×n соответственно, S - диагональная матрица, на диагонали которой расположены сингулярные числа матрицы A.
- svds(A) возвращает вектор, содержащий сингулярные числа матрицы A (A -вещественная матрица размером $m \times n$, где $m \ge n$).
- cholesky(A) возвращает нижнюю треугольную матрицу Т такую, что А=T·T^T (матрица А – симметрическая, при этом используется только верхняя треугольная часть этой матрицы).
- geninv (A) левая обратная к матрице А. L·A=E, где E единичная матрица размером n×n, L прямоугольная матрица размером n×m, A прямоугольная матрица размером m×n.
- eigenvals(M) возвращает вектор, содержащий собственные значения матрицы М.
- eigenvec(M, Z) возвращает нормализованный собственный вектор, соответствующий собственному значению Z квадратной матрицы M.
- eigenvecs(M) возвращает матрицу, столбцами которой являются нормализованные собственные векторы матрицы М (порядок расположения собственных векторов соответствует порядку расположения собственных значений, возвращаемых функцией eigenvals).
- genvals(M, N) возвращает вектор v обобщенных собственных значений, соответствующих решению уравнения M·x=v_i·N·x (М и N квадратные вещественные матрицы одинаковой размерности).
- genvecs(M, N) возвращает матрицу, содержащую нормированные обобщенные собственные векторы, соответствующие обобщенным собственным значениям, возвращаемым в векторе v функцией genvals.
- Isolve(A, B) возвращает вектор X, который является решением системы линейных алгебраических уравнений вида A·X=B, где A матрица размерности n×n, а B –вектор-столбец с n элементами.

4. Графические возможности MathCAD

Для создания графиков в системе MathCAD имеется программный графический процессор. Большинство параметров графического процессора, необходимых для построения графиков, задается автоматически. Поэтому для построения графиков того или иного вида достаточно задать тип графика.

4.1. Подменю <u>G</u>raphics главного меню

В опции <u>Graphics</u> (Графика) главного меню содержится восемь команд для задания графиков различных типов:

Create X-<u>Y</u> Plot *ⓐ* (Декартов график) – создать шаблон графика в декартовых координатах;

Create Polar Plot (Полярный график [Ctrl]+7) – создать шаблон графика в полярных координатах;

Create Surface Plot (График поверхности [Ctrl]+2) – создать шаблон для построения трехмерного графика;

Create Contour Plot (Карта линий уровня [Ctrl]+5) – создать шаблон для контурного графика;

Create 3D <u>Scatter Plot</u> (Точечный график) – создать шаблон для графика точек в трехмерном пространстве;

Create Vector <u>Field Plot</u> (Векторное поле) – создать шаблон для графика векторного поля на плоскости;

Create 3D <u>Bar Chart (Трехмерная гистограмма)</u> –создать шаблон для создания гистограммы в трехмерном пространстве;

Create Picture (Изображение) – создать шаблон области для импорта рисунка;

<u>X-Y Plot Format...(Формат Декартовых графиков...)</u> – изменить формат графика в декартовых координатах;

Polar Plot Format...(Формат полярных графиков...) – изменить формат графика в полярных координатах;

3<u>D</u> Plot Format...(Формат графика поверхности...) – изменить формат трехмерного графика;

Picture Format...(Формат изображения...) – изменить формат области для импорта рисунка.
4.2. Графики в декартовых координатах

Чтобы создать график, необходимо:

- установить курсор в том месте, где нужно создать график;
- выбрать команду Create X-Y Plot (Создать Декартов график) из опции <u>Graphics</u> (Графики) главного меню или нажать соответствующую кнопку на наборной панели графиков.

MathCAD создаст пустой шаблон графика с шестью полями для ввода, по три на каждой оси (см. рис. 4.1).

Поле в середине оси абсцисс предназначено для независимой переменной графика. В данное поле записывается дискретная переменная, переменная с



Рис. 4.1. Шаблон графика

индексом или любое выражение, содержащее дискретную переменную. Поле в середине оси ординат содержит выражение, график которого нужно построить. В поле ЭТО вводится дискретная переменная, переменная с индексом, или выражение, содержащее дискретную переменную (например, функции ИМЯ параметрами), находящуюся на оси абсцисс. Другие четыре пустые поля могут использоваться для непосредственной установки граничных значений на осях координат. В общем случае для

построения графика необходимо указать два набора данных: для оси абсцисс и для оси ординат. Эти наборы могут быть получены различными способами, например:

- заданием дискретной переменной величины в определенном диапазоне;
- вычислением какой-либо функции на некотором наборе данных;
- вычислением или заданием матрицы (вектора);
- считыванием матрицы (вектора) из текстового файла с помощью функции READPRN и т.д.

Можно начертить несколько кривых на одном чертеже. График может содержать несколько выражений по оси ординат, зависящих от одного выражения по оси абсцисс, или несколько выражений по оси ординат, согласованных с соответствующими выражениями по оси абсцисс. В первом случае в поле на оси ординат вводятся через запятую несколько выражений относительно одной переменной величины, которая указывается в поле на оси абсцисс. Во втором случае нужно ввести два или более выражения, отделяемых запятыми, на оси ординат и то же самое количество выражений на оси абсцисс. МаthCAD согласует выражения попарно — первое выражение на оси ординат с первым на оси абсцисс и т.д. Каждая согласованная пара должна использовать

одну дискретную переменную. Можно построить до 16 кривых в первом случае, и до 10 — во втором. Примеры изображения графиков показаны на рис. 4.2.



Рис. 4.2. Примеры графиков в MathCAD

Если выражение принимает комплексные значения, MathCAD отображает только вещественную часть, а мнимая часть игнорируется. При этом сообщений об ошибках отображаться не будет.

Для изменения формата графика нужно выбрать команду X-Y Plot Format..(Формат декартовых графиков...) или щелкнуть мышью в области графика для выделения его синей рамкой. В последнем случае меню Graphics (Графики) заменится на меню X-Y Plot (Декартов график). Затем нужно выбрать пункт Format... (Формат...) из меню X-Y Plot (Декартов график). В обоих случаях появится диалоговое окно Formatting Currently Selected X-Y Plot (Форматирование текущего X-Y графика). Это диалоговое окно появится и после того, как дважды щелкнуть мышью в области графика.

Диалоговое окно имеет панельный переключатель на четыре позиции:

X-Y Axes (X-Y Оси) – управление опциями осей;

Traces (Графики) – управление линиями графика;

Labels (Надписи) – управление надписями;

Defaults (По умолчанию) – задание опций по умолчанию.

При выборе панели **X-Y Axes (X-Y Ocu)** появится страница настройки параметров осей, вид которой показан на рис. 4.3. Каждая ось имеет следующие связанные с ней установки:

Log Scale – устанавливает логарифмический масштаб по выбранной оси, при этом границы по оси должны быть положительными числами;

<u>Grid Lines</u> – деления на выбранной оси заменяются линиями сетки;

Numbered – на выбранной оси у отметок проставляются числовые значения;

Autoscale – определяет способ автоматической установки границ на осях координат, если они не были определены вручную. Если этот флажок сброшен, MathCAD устанавливает границу на оси по соответствующему предельному значению данных, в противном случае в качестве границы на оси берется значение, соответствующее последующей за предельным значением данных осевой метке;

Show Markers – позволяет добавлять к графику фоновые линии;

Auto G<u>r</u>id – при установке этого флажка MathCAD автоматически выбирает число интервалов сетки, созданных отметками или линиями сетки на осях, в противном случае, можно установить число интервалов сетки, указав в поле No. o<u>f</u> Grids число от 2 до 99. Поле для ввода количества интервалов сетки доступно только в том случае если не установлен логарифмический масштаб.

Formetting Currently Selected X4Y Plot
X-Y Axes Traces Labels Defaults
X-Axis: 🗍 Log Scale Y-Axis: 🗍 L <u>og</u> Scale
🗇 <u>G</u> rid Lines 🗍 G <u>r</u> id Lines
⊠ N <u>u</u> mbered ⊡ Num <u>b</u> ered
🗹 Au <u>t</u> oscale 🗹 Auto <u>s</u> cale
🗇 S <u>h</u> ow Markers 👘 Sho <u>w</u> Markers
🗹 Auto G <u>r</u> id 🗹 Auto Gr <u>i</u> d
2 No. of Grids 2 No. of Gri <u>d</u> s
Axes Style Boxed Q Crossed Q None
OK Cancel <u>A</u> pply <u>H</u> elp

Рис. 4.3. Страница настройки параметров координатных осей

Страница X-Y Axes содержит три кнопки в группе Axes Style. Эти кнопки определяют стиль отображения координатных осей. Кнопка **Boxed** окружает график рамкой, кнопка **Crossed** показывает оси, пересекающиеся в центре графика. При выборе кнопки **None** оси не будут отображаться на графике.

Отдельные кривые на графике могут быть переформатированы с использованием страницы **Traces (Графики)** того же диалогового окна **Formatting Currently Selected X-Y Plot (Форматирование текущего X-Y графика)**. Эта страница управляет отображением линий, которыми строится график. Вид страницы настройки параметров кривых показан на рис. 4.4.

مسطلام كسطال هوايوريون الله المن الله الله الله الله الله الله الله الل			
Legend Label Symbol Line Color Type Weight Trace 1 none solid red Threes 1 F trace 2 none dot blu lines 1			
trace 3 none dash grn lines 1 trace 4 none dadot mag lines 1			
trace 1 none 🕞 solid 🔽 red 🔽 lines 🔽 1 🔽			
☐ Hide <u>A</u> rguments			
OK Cancel <u>A</u> pply <u>H</u> elp			

Рис. 4.4. Страница настройки параметров кривых

Так как на одном чертеже может располагаться до шестнадцати различных кривых, то каждой кривой соответствует строка в прокручивающемся списке. Каждая строка имеет шесть полей:

Legend Label – поле для ввода названия кривой, которое появляется под чертежом вместе с образцом линии;

<u>Symbol</u> – установка символа отметки точек графика (**none**–без отметки, × , + , □ , ◊ или **O**);

Li<u>n</u>e – указывает тип линии (none – ничего, solid – сплошная, dash – пунктирная, dadot – штрихпунктирная);

<u>Color</u> – определяет цвет кривой (red – красный, blu – синий, grn – зеленый, mag – фиолетовый. cya – голубой, brn – коричневый, blk – черный, wht – белый);

<u>Т</u>уре – управляет типом графика (line – линия, points – точки, err – интервал ошибок, bar – гистограмма, step – ступенчатая кривая, draw – рисунок);

<u>Weight</u> – управляет толщиной линии. Значения варьируются от 1 до 9, также можно указать толщину линии, равную Р, что означает толщину в один пиксель *устройства*, т.е. экрана или принтера. Принтер с высоким разрешением будет отображать кривую с такой установкой как очень тонкую линию.

Кроме прокручивающегося списка страница **Traces** имеет два переключателя: **Hide** <u>Arguments</u> (Скрыть переменные) и <u>H</u>ide Legend (Скрыть имена кривых).

Страница Label (Надписи) появляется, если уже создан текущий график. Надписи задаются в окнах:

Title (Заголовок) – титульная надпись;

<u>X</u>-Axis (X-Ось) – надпись по оси X;

<u>Y</u>-Axis (Y-Ось) – надпись по оси Y.

В разделе Title (Заголовок) содержатся опции: <u>Above (Сверху)</u>, <u>Below</u> (Снизу) и <u>Show Title (Показать заголовок)</u>, которая позволяет отображать титульную надпись.

Страница Defaults (По умолчанию) служит для установки опций <u>Change to Defaults (Вернуть значения по умолчанию)</u> и <u>Use for Defaults</u> (Использовать для значений). Установленные и зафиксированные опции используются в дальнейшем при построении графиков функций одной переменной.

Для изменения размеров графика необходимо сначала выделить его пунктирной рамкой. Для этого нужно щелкнуть мышью вне области графика, затем переместить курсор мыши в область графика, удерживая при этом



Рис. 4.5. Окно Х-Ү Zoom

клавишу мыши и отпустить клавишу. Далее нужно переместить курсор мыши к правой или нижней стороне появившегося прямоугольника. Курсор при ЭТОМ изменится на двойную стрелку. Нажав и клавишу мыши, удерживая нужно переместить мышь по направлению, в котором требуется изменить размер графика. Как только графическая область лостигнет нужного размера, можно отпустить кнопку мыши. Для отмены

выделения графика необходимо щелкнуть мышью вне области графика.

МathCAD позволяет увеличить отдельную область графика. Для этого необходимо поместить курсор в область графика и нажать левую клавишу мыши, чтобы заключить график в синий выделяющий прямоугольник. При этом меню **Graphics (Графика)** заменится на меню **X-Y Plot (Декартов график)**, из которого нужно выбрать пункт **Zoom...** (Окно...). Появится диалоговое окно **X-Y Zoom** (см. рис. 4.5).

Фрагмент графика, подлежащий увеличению, необходимо заключить в пунктирный выделяющий прямоугольник. Для этого нужно поместить курсор мыши внутрь области графика и, нажав и удерживая левую кнопку, переместить мышь. Появится пунктирный прямоугольник. Когда вся область, которую нужно увеличить попадет в этот прямоугольник, отпустите левую кнопку мыши. Координаты выбранной области отображаются в полях **Міп** и **Мах** диалогового окна **Х-Ү Zoom**. Кнопки окна выполняют следующие действия:

Zoom – увеличивает выбранную область и не закрывает окно **X-Y Zoom**;

<u>U</u>nzoom – отменяет увеличение области графика;

<u>Full View</u> – если у исходного графика уже был изменен масштаб, то нажатие этой кнопки приводит к восстановлению *первоначального* масштаба графика;

<u>Accept</u> – "принимает" сделанные изменения масштаба графика и закрывает окно X-Y Zoom;

<u>Cancel</u> – "отменяет" изменения масштаба и закрывает окно X-Y Zoom.

Кроме того, MathCAD позволяет считывать координаты каждой *дискретной* точки графика. Для этого необходимо щелкнуть мышью на



Рис. 4.6. Окно Х-Ү Тгасе

чтобы графике, выделить его синей рамкой. При ЭТОМ меню Graphics (Графика) заменится на меню X-Y Plot (Декартов график), из которого нужно выбрать пункт Тгасе...(Отслеживать...). На экране появится диалоговое окно Х-Ү Тгасе (см. рис. 4.6). Необходимо установить курсор мыши примерно в ту координаты точку графика, которой требуется считать, и нажать кнопку

мыши. Для перемещения по точкам кривой используются клавиши [\leftarrow] и [\rightarrow], а для выбора кривой – [\uparrow] и [\downarrow]. Применение мыши целесообразно при грубой установке маркера (перекрещивающихся линий), а для более точной установки удобнее использовать клавиатуру. В полях X-Value и Y-Value отображаются координаты выбранной точки. Чтобы скопировать координаты в буфер обмена, достаточно нажать на кнопки Сору <u>X</u> и Сору <u>Y</u>. Перекрещивающиеся

пунктирные линии остаются до тех пор, пока не щелкнуть мышью вне области графика.

4.3. Графики в полярных координатах

Чтобы создать график в полярных координатах, необходимо:

- установить курсор в том месте, где нужно создать график;
- выбрать команду Create Polar Plot (Создать Полярный график) из меню <u>Graphics</u> (Графики) или нажать соответствующую кнопку на наборной панели графики.

MathCAD создает пустой шаблон графика с четырьмя полями для ввода, по три на каждой оси (см. рис. 4.7).



Рис. 4.7. Шаблон графика

Поле ввода внизу предназначено для угловой переменной графика. Введите туда дискретную переменную любое или выражение, содержащее дискретную переменную. Поле ввода, расположенное слева, должно содержать выражение для радиуса. Два справа поля ввода верхнего и нижнего предназначены для граничных значений радиуса. MathCAD заполняет эти поля по умолчанию. При необходимости можно изменить значения в этих полях.

В MathCAD графики в полярных координатах строятся путем пересчета модуля радиус-вектора r и его аргумента θ в декартовы координаты x и y с использованием стандартных преобразований ($x = r \cdot cos(\theta)$ и $y = r \cdot sin(\theta)$).

Точно так же, как можно построить несколько графиков на одном чертеже в декартовых координатах, можно начертить несколько графиков на одном и том же чертеже в полярных координатах.

Для изменения формата графика нужно выбрать команду <u>Polar Plot</u> Format...(Формат полярных графиков...) или щелкнуть мышью в области графика для выделения его синей рамкой. В последнем случае меню <u>G</u>raphics (Графики) заменится на меню <u>Polar Plot (Полярный график)</u>. Затем нужно выбрать команду <u>Format...</u> (Формат...) из меню <u>Polar Plot (Полярный</u> график). В обоих случаях появится диалоговое окно Formatting Currently Selected Polar Plot (Форматирование текущего полярного графика). Это диалоговое окно появится и после того, как дважды щелкнуть мышью в области графика. Диалоговое окно имеет панельный переключатель на четыре позиции:

Polar Axes (Полярные Оси) – управление опциями осей;

Traces (Графики) – управление линиями графика;

Labels (Надписи) – управление надписями;

Defaults (По умолчанию) – задание опций по умолчанию.

При выборе панели **Polar Axes (Полярные Оси)** появляется страница настройки параметров координатных осей, вид которой показан на рис. 4.8.

Formatting	<u>Currently Selecter</u>	i Poler Plot		×
Polar A	xes Traces	Labels	Defaults	
Radial:	🔟 Log Scale	Angula	ar:	
	🔟 <u>G</u> rid Lines		🔟 G <u>r</u> id Line	es
	🗹 N <u>u</u> mbered		🗹 Nu <u>m</u> bero	ed
	🔟 <u>S</u> how Mark	ers		
	🗹 <u>A</u> uto Grid		🗹 Au <u>t</u> o Grie	t l
	4 N <u>o</u> . of Gr	ids	12 No. of	Grids
	Axes Style=		rossed () <u>N</u> or	ie
` 0	K Ca	ancel	ĕbbîà	<u>H</u> elp

Рис. 4.8. Страница настройки параметров координатных осей для графика в полярных координатах

Каждая ось имеет следующие связанные с ней установки:

Log Scale – устанавливает логарифмический масштаб радиальной оси, при этом границы по оси должны быть положительными числами. Эта установка доступна только для *радиальной* оси;

<u>Grid Lines</u> – заменяются деления на выбранной оси линиями сетки (радиальные линии — окружности; угловые — лучи);

Numbered – на выбранной оси у отметок проставляются числовые значения;

Show Markers – добавляются к графику фоновые линии;

Auto G<u>r</u>id – автоматически выбирается число интервалов сетки. В противном случае можно установить число интервалов сетки, указав в поле No. o<u>f</u> Grids число от 2 до 99. Поле для ввода числа интервалов сетки доступно только в том случае, если не установлен логарифмический масштаб.

Также страница Polar Axes содержит три кнопки в группе Axes Style. Эти кнопки определяют стиль, в котором будут отображаться координатные оси. При выборе кнопки Perimeter (Обрамление) график обводится окружностью, кнопки Crossed (Визир) – появляются оси, пересекающиеся в центре графика, кнопки None (Ничего) – ничего не отображается на графике.

Traces (Графики), Label (Надписи), Defaults Страницы (По умолчанию) диалогового окна Formatting Currently Selected Polar Plot (Форматирование текущего полярного графика) аналогичны соответствующим страницам диалогового окна Formatting Currently Selected X-Y Plot (Форматирование текущего X-Y графика), но страница Label (Надписи) позволяет устанавливать только титульную надпись. Изменение представления графиков, их масштабирование, считывание координат точек графиков, аналогичны соответствующим возможностям по обработке графиков в декартовых координатах.

4.4. Графики поверхностей

В рабочие документы MathCAD можно включать трехмерные графики: поверхности, карты линий уровня трехмерных поверхностей, трехмерные гистограммы, векторные и точечные графики.

Чтобы создать поверхность, необходимо:

- определить матрицу значений, которую требуется отобразить графически. MathCAD использует номера строк и столбцов в качестве координат по осям X и Y, а значения элементов матрицы в качестве координат по оси Z;
- выбрать команду Create Surface Plot (График поверхности) в опции <u>Graphics</u> (Графика) главного меню или нажать соответствующую кнопку на наборной панели графиков. MathCAD отобразит рамку с одним полем ввода (графическую область поверхности);
- указать имя матрицы в поле ввода;
- нажать клавишу [F9] или щелкнуть мышью вне графической области.

Каждый элемент матрицы представляется как точка на определенной высоте, пропорциональной значению этого элемента. Координаты проекции точки на плоскость (X-Y) являются соответственно номерами строки и столбца матрицы. Точки соединяются линиями, которые и образуют поверхность.

МathCAD предоставляет различные возможности по изменению внешнего вида графика поверхности. Чтобы ими воспользоваться, нужно выбрать команду **3D Plot Format...** (Формат <u>3D</u>-графика...) из опции **Graphics (Графика)** главного меню или дважды щелкнуть мышью в области графика. Появится диалоговое окно **3D Plot Format**, которое имеет панельный переключатель на четыре позиции:

View (Вид) – установка опций обзора;

Axes (Оси) – установка опций представления осей;

Color & Lines (Цвет и линии) – установка цвета и типа линий;

Title (Заголовок) – установка титульной надписи.

Страница View (Вид) диалогового окна 3D Plot Format представлена на рис. 4.9. Эта страница содержит пять групп контролей (полей редактирования, переключателей, флажков и кнопок): View (Вид), Axes (Оси), Back Planes (Грани), Display As (Отобразить как) и Frames (Рамки).

3D Plot Format	X
View Axes Color & Lines	Title)
View: <u>Rotation: 10</u> 0 to 360° <u>Tilt: 35</u> 0 to 360° <u>V</u> ert. Scale: 100 1 to 100%	Axes: Back Planes: @ Perimeter J Show @ Corner Fill Color @ None Edge Color
Display As: © Surface Plot © Data Poin © Contour Plot © 3D Bar Ch	ts. yant ☐ Show Box
OKCano	:el <u>≜pp</u> ly <u>H</u> elp

Рис. 4.9. Окно настройки параметров трехмерного графика

Группа View (Вид) содержит опции:

<u>Rotation (Вращение)</u> – задает угол поворота;

<u>Tilt (Наклон)</u> – задает угол наклона;

Vert. Scale (Вертикальный Масштаб) – задает масштаб по оси Z.

Группа Axes (Оси) содержит опции для задания типа отображения осей:

Perimeter (Периметр) – размещает координатные оси по периметру;

Corner (Репер) – размещает координатные оси в углу;

<u>None (Ничего)</u> – запрещает отображение осей.

В группу Back Planes (Грани) входит флажок Show (Показать), установка которого разрешает отображение координатных плоскостей, и две кнопки <u>Fill Color...(Закрасить...)</u> и Edge Color...(Цвет ребер...), вызывающие диалоговое окно настройки цвета плоскостей и линий их пересечения соответственно.

Группа **Display As (Отобразить как)** содержит четыре переключателя, определяющие тип трехмерного графика:

<u>Surface Plot</u> – представляет график в виде поверхности;

<u>Contour Plot</u> – вычерчивает карту линий уровня поверхности;

Data <u>Points</u> – отображает точечный график;

3D <u>**B**</u>ar Chart – строит трехмерную гистограмму.

Заметим, что некоторые группы контролей на различных страницах диалогового окна **3D Plot Format** могут быть недоступны, так как зависят от положения переключателей группы **Display As** (Отобразить как). Например, при выборе переключателя **Contour Plot** остается доступной только группа **Display As**.

Группа **Frames (Рамки)** содержит опции для установки ограничивающих элементов:

Show Border (Рамка) – заключает графическую область а рамку;

Show Box (Ящик) — заключает график в параллелепипед.

Страница Axes (Оси) диалогового окна 3D Plot Format изображена на рис. 4.10. Она содержит три группы контролей <u>X</u>-Axis, <u>Y</u>-Axis и <u>Z</u>-Axis для настройки осей координат. Все группы содержат одинаковый набор контролей, действие которых распространяется на одну конкретную ось. Все контроли на странице Axes (Оси) становятся недоступными, если на странице View (Вид) в группе Axes (Оси) установлен переключатель None (Ничего). Каждая группа содержит следующие опции:

Grid Lines (Линии сетки) – выводит линии сетки на соответствующей оси;

Numbered (Пронумеровать) – отображает риски с числами на конкретной оси;

Auto Grid (Автосетка) – автоматический выбор числа линий сетки на координатной оси. В противном случае необходимо ввести их число в поле редактирования <u>No. of Grids (Число линий)</u>;

<u>No. of Grids (Число линий)</u> – задание числа линий сетки (доступно только при сброшенном флажке Auto Grid);

<u>Min. Val. – минимальное значение соответствующей координаты;</u>

Max. <u>Val.</u> – максимальное значение соответствующей координаты;

Autoscale (Автомасштаб) – автоматическое деление осей.

Поля для осей X и Y доступны только в том случае, если на графике отображается параметрическая поверхность, иначе и эти поля, и соответствующий им флажок Autoscale недоступны. Если же флажок Autoscale доступен, то для получения доступа к полям Min. Val. и Max. Val. необходимо его сбросить.

2	D Plot Famel		×
	View Axes Color	& Lines Title	
	X-Axis	Y-Axis	Z-Axis
	🗇 Grid Lin <u>e</u> s	🔟 Grid Lin <u>e</u> s	🔟 Grid Lin <u>e</u> s
	🗹 Num <u>b</u> ered	🗹 Num <u>b</u> ered	🗹 Num <u>b</u> ered
	🗹 Auto Gri <u>d</u>	🗹 Auto Gri <u>d</u>	🗹 Auto Gri <u>d</u>
	4 No. of Grids	4 No. of Grids	4 No. of Grids.
	-1. <u>Min</u> , Val,	-1. Min. Val.	-1. Min. Val.
	1 Ma <u>x.</u> Val.	1. Max. Val.	1. <u>Max.</u> Va <u>l</u> .
	🗵 <u>Autoscale</u>	🗵 A <u>utoscale</u>	🗹 Autoscale
	ОК	Cancel	<u>≜pply</u> <u>H</u> elp

Рис. 4.10. Страница настройки параметров координатных осей для поверхности

На странице Color & Lines (Цвета и линии) диалогового окна 3D Plot Format при отображении графика в виде поверхности находятся две доступные группы контролей: Shading (Окраска) и Fill Style (Стиль заполнения). В группе Shading (Окраска) находятся три опции: Color (Цветная), Grayscale (Черно-белая) и None (Нет), которые задают способ окраски поверхности. По умолчанию установлена опция None. В группе Fill Style (Стиль заполнения) находятся две опции: Hidden Lines (Невидимые линии) и Patch Plot (Площадки). При установке первого флажка поверхность становится непрозрачной, т.е. части поверхности, которые должны быть закрыты другими выступающими частями, не отображаются. Если установлен второй флажок, то отображается каждый элементарный фрагмент поверхности В виде прямоугольника. Если установлен переключатель Color (цветная) окраска поверхности, то красный цвет характеризует максимум по оси Z, а фиолетовый - минимум. При Grayscale (Черно-белая) окраске белый цвет характеризует максимум, а черный – минимум. Если используется возможность построения фигур с интерполяцией Interpolated Mesh (Сетка интерполяции), то страница Color & Lines (Цвета и линии) позволяет установить No. of Rows (Число строк) и No. of Columns (Число столбцов) для узловых точек фигуры. Группа Contours (Уровни) задает параметры построения контурных графиков: Contour Lines (Линии уровня), Numbered (Пронумеровать), Auto Contour (Автолинии), No. of Contours (Число линий).

Страница Title (Заголовок) диалогового окна 3D Plot Format позволяет ввести титульную надпись и активизировать опции: Above (Сверху), Below (Снизу) и Show (Показать заголовок).

Можно построить параметрическую поверхность. Чтобы использовать графические возможности MathCAD для отображения параметрической поверхности, необходимо:

- заполнить данными три одинаковые по размерности матрицы;
- напечатать их имена в поле ввода графической области поверхности через запятую;
- нажать клавишу [F9] или щелкнуть мышью вне графической области.



Рис. 4.11. Пример параметрической поверхности

MathCAD интерпретирует три введенные матрицы как *x*, *y* и *z* координаты точек поверхности. Необходимо иметь в виду, что нельзя преобразовать параметрическую поверхность в другой тип трехмерного графика.

Построение других типов графиков и их форматирование подробно не рассматриваются, так как их отличие не столь значительно.

5. Программирование в системе MathCAD

5.1. Введение в программирование

Начиная с версии 6.0, в MathCAD появилась возможность создания программных модулей. Программы MathCAD содержат конструкции, во многом подобные конструкциям других языков: условные операторы, циклы, подпрограммы, рекурсии и т.д.

Программа MathCAD есть частный случай выражения MathCAD. Подобно любому выражению программа возвращает значение, если за ней следует знак равенства. Точно так же, как переменную или функцию можно определить через выражение, их можно определить и с помощью программы.

Главным различием между программой и выражением является способ задания вычислений. При использовании выражения алгоритм получения ответа должен быть описан одним оператором. В программе может быть использовано столько операторов, сколько требуется. Можно рассматривать программу как «составное выражение».

Программные средства MathCAD сосредоточены на наборной панели программных структур, которая содержит семь элементов:

Add Line	 создает и при необходимости расширяет вертикальную линию, справа от которой в полях ввода задается тело программы;
←	– оператор локального присваивания;
if	 оператор проверки условного выражения;
otherwise	- оператор иначе (применяется вместе с if);
while	– оператор задания цикла типа пока ;
for	– оператор задания цикла с заранее известным числом шагов;
break	 оператор прерывания программы или цикла.
-	

Заметим, что ввод операторов программирования производится только с помощью панели программных структур.

5.2. Этапы написания программы

Программа, в сущности, является функцией, описанной с применением программных средств. Обычно программному модулю присваивается имя со списком переменных, после которого идет оператор :=. Переменные в списке являются формальными параметрами программы, и их значения, т.е. фактические параметры, нужно обязательно указывать при вызове этой программы. Кроме того, в программный модуль могут быть переданы данные через локальные (объявленные в документе до программного модуля) или глобальные (заданные с помощью знака =) переменные.

Для написания простейшей программы, например, вычисления функции f(x,w)=log(x+w), необходимо:

- ввести левую часть определения функции и знак := ;
- на панели программных структур нажать кнопку Add Line, после чего появится вертикальная линия с двумя полями ввода;
- в верхнем поле ввода напечатать z;
- нажать кнопку ← ;
- нижнее поле ввода предназначено для возвращаемого значения log(z).

На рис. 5.1 приведены основные этапы написания этой программы.

f(x,w) =	f(x,w) :=	$Z \leftarrow X + W$	f(x,w) =	$Z \leftarrow X + W$	f(2,3) = 0.69897
		•		log(z)	

Рис. 5.1. Этапь	и написания	программы
-----------------	-------------	-----------

Заметим, что переменная z не определена вне программы, ее определение является локальным и действует только внутри программы.

Программа может состоять из любого числа операторов. Чтобы расширить тело программы, нужно щелкнуть по кнопке Add Line, при этом появится свободное поле ввода. Чтобы удалить позицию ввода, ее нужно выделить, заключив в синюю рамку, и нажать клавишу [Del].

Для вывода из программы вычисленного значения переменной, вектора или матрицы используется знак = , причем возвращается значение последней строки программы.

5.3. Составной оператор ...if... ...otherwise

При необходимости создания разветвления в программе применяется составной оператор ...if... ...otherwise. Оператор if может применяться как самостоятельно, так и в комбинации с оператором otherwise. Чтобы ввести оператор if, необходимо нажать соответствующую кнопку наборной панели программирования. Слева и справа от if появятся поля ввода. Правое поле предназначено для условного выражения, а левое – для значения, которое будет иметь выражение, если условие в правом поле истинно. При необходимости в нижнее поле ввода щелчком мыши по соответствующей кнопке вводится otherwise. В поле слева от otherwise записывается значение, которое программа должна возвратить, если логическое выражение справа от оператора if ложно. Заметим, что оператор ...if... ...otherwise можно использовать и тогда, когда число вариантов больше двух. На рис. 5.2 показаны примеры работы оператора ...if... ...otherwise.



Рис. 5.2. Примеры использования оператора ...if... ...otherwise

В системе MathCAD не предусмотрено логических **и**, **или**, как в других языках программирования. Для замены этих логических функций можно применять соответственно арифметические операции умножения и сложения,

но, при этом, необходимо учитывать, что в MathCAD выражение истинно, когда его значение не равно нулю, и ложно, когда значение равно нулю. При необходимости проверки на равенство нужно использовать полужирный знак равенства, который можно ввести с панели операторов отношения или нажатием клавиш [Ctrl]+[=].

5.4. Циклы в MathCAD

Имеется два вида циклов:

- цикл типа for, который используется, если заранее известно число шагов (итераций цикла);
- цикл типа while, который применяется, если число шагов не определено, а выход из цикла должен быть произведен только при прекращении выполнения некоторого условия.

Для создания цикла типа **for**, необходимо:

- щелкнуть по кнопке **for** на панели программных структур;
- напечатать в поле ввода слева от знака ∈ имя переменной цикла;
- ввести в поле справа от знака ∈ диапазон значений, в котором должна изменяться переменная цикла. Форма задания диапазона значений аналогична форме задания дискретного аргумента, т.е. первым указывается начальное значение переменной, затем, после запятой, второе значение и последним, после многоточия, конечное значение. Шаг цикла равен разности между вторым и первым значениями переменной цикла. По умолчанию шаг считается равным 1. Выражение справа от знака ∈ может быть также списком скаляров, векторов и диапазонов, разделенных запятыми. При этом диапазоны необходимо заключать в скобки;
- в нижнее поле ввода записать тело цикла.

Для создания цикла типа **while** необходимо:

- щелкнуть по кнопке while на панели программных структур;
- напечатать условие выполнения цикла в верхнем поле ввода;
- в нижнем поле ввода записать тело цикла.

Иногда возникает необходимость прерывания работы цикла до его завершения. Сделать это позволяет оператор **break**, прерывающий выполнение того цикла, в теле которого он находится. Если оператор **break** находится вне тела цикла, то он прерывает выполнение самой программы. Часто оператор **break** применяется в комбинации с оператором ...if... ...**otherwise**. В этом случае чаще всего применяются следующие формы записи операторов:

- break if ...;
- ... if ...

break otherwise.

Примеры использования циклов и оператора **break** приведены на рис. 5.3 и рис. 5.4.

Вложенные циклы		Цикл for со сложным списком
Ident(n) :=	for $i \in 0 n - 1$ for $j \in 0 n - 1$ $M_{i,j} \leftarrow 1$ if $i \bullet j$ $M_{i,j} \leftarrow 0$ otherwise M	параметров $\gamma := (8 \ 7 \ 6)^{T}$ $f(v) := m \leftarrow 0$ for $x \in (31), (45), v, 9$ $ V_{m} \leftarrow x$ $m \leftarrow m + 1$
Ident(4) =	$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	V f(γ) ^T = (3 2 1 4 5 8 7 6 9)

Рис.	5.3.	Примеры	использования	циклов	типа	for
------	------	---------	---------------	--------	------	-----

J(h) :=	i← 0	ſ	2.66667	
	while $i \le 10$		5.20833	
	$x_i \leftarrow 2 + i \cdot h$	J(0.5) =	9	
	ſ*X;		14.29167	
	$\left S_{i} \leftarrow \right _{0}^{1} t^{2} dt$		21.33333	
	break if $S_i \ge 20$			
	i←i + 1			
	S			



5.5. Рекурсивные функции и подпрограммы

Рекурсивная функция – это функция, внутри которой находится вызов самой этой функции. В некоторых ситуациях, например, при программировании многоуровневых вычислений, рекурсия является достаточно полезной. На рис. 5.5 приведен пример рекурсивной функции, вычисляющей n!.

factorial(n) := $\begin{vmatrix} 1 \\ n \cdot factorial(n - 1) & \text{if } n \ge 1 \end{vmatrix}$ factorial(0) = 1 factorial(6) = 720

Рис. 5.5. Пример программы, вычисляющей п!

Подпрограмма – это логически завершенный программный модуль, который может быть вызван из другой функции необходимое число раз. В системе MathCAD на взаимное расположение подпрограммы и функции, ее вызывающей, накладываются следующие условия:

- если подпрограмма и функция определяются с помощью одинаковых знаков присваивания (:= или =) или подпрограмма определяется знаком := , а функция знаком = , то подпрограмма обязательно должна располагаться в рабочем документе до функции;
- если подпрограмма определена с помощью знака глобального присваивания (≡), а функция – с помощью знака локального присваивания (:=), то подпрограмма может находиться в любом месте рабочего документа по отношению к вызывающей ее функции.

На рис. 5.6 приведен пример вызова функцией подпрограммы **factorial(n)**. Функция **floor**, используемая в программе, является встроенной функцией MathCAD и выполняет округление своего аргумента до наибольшего целого числа.

$$factorial(n) := \begin{vmatrix} 1 \\ n \cdot factorial(n-1) & \text{if } n \ge 1 \end{vmatrix}$$
$$f(x) := \begin{vmatrix} factorial(floor(x)) & \text{if } x \ge 0 \\ 0 & \text{otherwise} \end{vmatrix}$$



5.6. Возврат значений из программы

Из программы может быть выдано только одно выражение. Выходным выражением может быть либо матрица, либо вектор, либо скаляр. Это весьма неудобно, если результатом работы являются значения нескольких выражений. В этом случае необходимо организовать вектор, элементами которого могут быть переменные, матрицы и векторы одновременно, но при этом нужно запомнить позиции этих элементов (см. рис. 5.7).

F(M) :=	$\operatorname{ret}_0 \leftarrow M^{-1}$	
	$\operatorname{ret}_1 \leftarrow M $	Программа, вычисляющая обратную
	$ret_2 \leftarrow eigenvals(M)$	собственных значений исходной матрицы.
	ret	
1 5	5 2	
A := 6 3	$ \mathbf{F} = \mathbf{F}(\mathbf{A})$	Исходная матрица и вызов программы.
$\langle 2 \rangle$	(22)	
Result =	{3,3} -40	Зизиение переменной Result
Kesun –	{3,1}	эначение переменной Кезин.
	-0.3 0.5 -0.85 \	
$\text{Result}_0 =$	0.2 0 -0.1	Обратная матрица размерности 3х3.
	0.15 -0.25 0.675	
$\text{Result}_1 =$	-40	Значение определителя (скаляр).
	-2.74909	
$\text{Result}_2 =$	9.16077	Вектор из собственных значений.
	1.58832	

Рис. 5.7. Возврат из программы нескольких значений

6. Функции обработки наборов данных

6.1. Интерполяция и функция предсказания

Интерполяция — это вычисление значения функции, заданной на некоторой сетке значений, в точке, находящейся внутри сетки. В системе MathCAD интерполяция представлена линейными и кубическими сплайнами.

Линейное предсказание заключается в использовании значений данных, заданных на сетке значений, для предсказания значения за ее пределами. В MathCAD имеется одна функция, предназначенная для линейного предсказания.

6.1.1. Сплайны

При интерполяции линейными сплайнами MathCAD возвращает значение сплайна в заданной точке. Это выполняется функцией linterp(vx,vy,x), параметры которой имеют следующий смысл:

vx – вектор х-координат заданных точек, расположенных по возрастанию;

vy – вектор у-координат заданных точек (размерности векторов **vx** и **vy** должны совпадать);

х – координата точки, в которой вычисляется значение линейного сплайна.

На рис. 6.1 изображен линейный сплайн, построенный по точкам, первые координаты которых равны целым числам от 0 до 9, а вторые координаты – случайные числа, равномерно распределенные на интервале (0,1), полученные с помощью функции rnd(1). Кроме того, приведены значения этого сплайна в точках с координатами 1.15 и 7.69.



Рис. 6.1. Линейный сплайн

Если значение **x** лежит левее или правее первого или последнего элемента в **vx**, то функция возвращает экстраполированное значение сплайна, полученное продолжением соответствующего крайнего отрезка.

Кубическая сплайн-интерполяция осуществляется функцией interp (vs, vx, vy, x), параметры которой имеют следующий смысл:

vs – вектор вторых производных интерполируемой кривой в точках сетки, генерируемый на основе векторов данных **vx** и **vy** одной из функций **lspline(·)**, **pspline(·)** или **cspline(·)**;

vx – вектор х-координат заданных точек, расположенных по возрастанию;

vy – вектор у-координат заданных точек (размерности векторов **vx**, **vy** и **vs** должны совпадать);

х – координата точки, в которой вычисляется значение кубического сплайна.

Функции lspline(·), pspline(·) и cspline(·) предназначены для вычисления векторов vs вторых производных интерполируемой кривой в соответствии с граничными условиями, определенными внутри каждой из функций. При использовании функции lspline(vx,vy) интерполируемая кривая приближается к прямой линии в граничных точках; pspline(vx,vy) - интерполируемая кривая в граничных точках приближается к параболе; cspline(vx,vy) - интерполируемая кривая в граничных точках приближается к кубической параболе.



Рис. 6.2. Интерполяция кубическими сплайнами

Пример использования интерполирования кубическими сплайнами приведен на рис. 6.2. Интерполирование производится на интервале [-2;2]. За границами интервала возвращаются экстраполированные значения.

6.1.2. Линейное предсказание

Иногда необходимо оценить значения функции в точках, находящихся вне области заданных значений. В MathCAD есть функция **predict(·)**, которая позволяет это сделать. Функция использует линейный алгоритм предсказания. Этот алгоритм можно рассматривать как разновидность экстраполяции, но его нельзя путать с линейной или полиномиальной экстраполяцией.

Функция predict (v, m, n) возвращает n предсказанных значений, основанных на m последовательных значениях вектора данных v. Функция имеет следующие параметры:

v – вектор значений функции, заданный в равноотстоящих точках;

m – число последовательных значений функции, на основе которых осуществляется предсказание;

n – число предсказанных значений.

Пример функции линейного предсказания приведен на рис. 6.3.



Рис. 6.3. Линейное предсказание

6.2. Функции регрессии

Система MathCAD содержит ряд функций для вычисления регрессии. Функции отличаются друг от друга прежде всего типом кривой, которую они используют для аппроксимации данных.

6.2.1. Линейная регрессия

При использовании функции линейной регрессии возвращаются наклон и смещение прямой линии, которая наилучшим образом приближает исходные данные в смысле наименьших квадратов отклонений. Прямая линия строится в виде

 $y(x) = slope(vx, vy) \cdot x + intercept(vx, vy)$,

где функция **slope(vx,vy)** возвращает скаляр, определяющий наклон линии регрессии к оси абсцисс, а функция **intercept(vx,vy)** – скаляр, определяющий смещение по оси ординат. Функции **slope(vx,vy)** и **intercept(vx,vy)** имеют следующие параметры:

vж – вектор х-координат заданных точек;

vy – вектор у-координат заданных точек (размерности векторов **vx** и **vy** должны совпадать).

Пример применения функции линейной регрессии приведен на рис. 6.4.



Рис. 6.4. Линейная регрессия

6.2.2. Полиномиальная регрессия

Полиномиальная регрессия применяется в том случае, если исходные данные аппроксимируют полиномом некоторой степени.



Рис. 6.5. Полиномиальная регрессия

Полиномиальная регрессия осуществляется функцией interp(vs,vx,vy,x), которая имеет следующие параметры:

vs – вектор, вычисляемый на выбор двумя функциями: **regress** (**vx**, **vy**, **n**) – если используется один полином степени **n** для аппроксимации всех данных, при этом не рекомендуется использовать **n>4**; **loess** (**vx**, **vy**, **span**) – если используется набор полиномов второй степени, наилучшим образом приближающие определенные окрестности исходных данных. Значение параметра **span** определяет величину окрестности локального приближаются данные (хорошим считается значение **span**=0.75);

vx – вектор х-координат заданных точек;

vy – вектор у-координат заданных точек (размерности векторов **vx**, **vy** и **vs** должны совпадать);

х – координата точки, в которой вычисляется значение функции регрессии.

На рис. 6.5, где приведен пример полиномиальной регрессии. Показано, что при малом значении **span=0.1** достаточно хорошо отслеживаются случайные колебания значений функции, а при **span=0.75** кривая регрессии становится практически гладкой.

6.2.3. Обобщенная регрессия

Аппроксимирующая функция задается в виде линейной комбинации функций из вектор-функции **F**



Рис. 6.6. Обобщенная регрессия: функция linfit(·)

Коэффициенты этой линейной комбинации определяются методом наименьших квадратов функцией linfit(vx,vy,F), которая имеет следующие параметры:

vж – вектор х-координат заданных точек;

vy – вектор у-координат заданных точек;

F – вектор–функция, элементы которой используются для построения аппроксимирующей функции в виде линейной комбинации функций из **F** (размерности векторов **vx**, **vy** и **F** должны совпадать).

На рис. 6.6 приведен пример использования linfit(·).

Для построения аппроксимирующей функции используется также функция genfit(.).



Рис. 6.7. Обобщенная регрессия: функция genfit (·)

Функция **genfit(vx,vy,vg,F)** находит вектор неизвестных коэффициентов A_i для нелинейной функции **f(x,A_0,A_1,...,A_n)**, которая наилучшим образом приближает данные методом наименьших квадратов. Параметрами функции **genfit(.)** являются:

vx – вектор х-координат заданных точек;

vy – вектор у-координат заданных точек (**vx** и **vy** – векторы одинаковой размерности);

vg — вектор размерности n+1 начальных значений для A_i , так как для их нахождения решается система нелинейных уравнений;

F – вектор-функция размерности n+2, элементами которой являются функция **f** (·) и ее частные производные по A_i .

На рис. 6.7 приведен пример использования genfit(·) для построения аппроксимирующей функции.

6.3. Функции сглаживания

Сглаживание предполагает использование набора данных **vy1** (и возможно **vx**) для получения **нового набора значений vy2**, который будет являться более гладким, чем исходный набор **vy1**. В отличие от интерполяции и регрессии функции сглаживания не дают значений в промежуточных точках, т.к. они возвращают только набор значений и **не позволяют построить аппроксимирующую функцию**.

Функция **medsmooth** (·) — наиболее устойчивая из трех существующих, т.к. в меньшей степени подвержена влиянию зашумленных данных. Эта функция использует сглаживание с помощью скользящей медианы, вычисляет остатки, сглаживает их тем же способом и суммирует два полученных сглаженных вектора. Необходимо отметить, что, если **n** – ширина окна, по которому происходит сглаживание, то функция **medsmooth** (·) будет оставлять первые и последние $\frac{n-1}{2}$ точки неизменными. На практике ширина окна сглаживания **n** должна быть мала по сравнению с размером массива данных.

Функция **ksmooth**(·) использует гауссово ядро, чтобы вычислить локально взвешенные средние значения исходного вектора **vy1**. Эта функция наиболее полезна, когда данные взяты в точках, отделенных друг от друга приблизительно равными интервалами.

Для каждого значения $vy1_i$ в n-мерном векторе vy1 функция ksmooth (·) возвращает значение $vy2_i$, определяемое формулой

64

$$vy2_{i} = \frac{\sum_{j=1}^{n} K\left(\frac{vx_{i} - vx_{j}}{b}\right) \cdot vy1_{j}}{\sum_{j=1}^{n} K\left(\frac{vx_{i} - vx_{j}}{b}\right)},$$

где

$$\mathbf{K}(\mathbf{t}) = \frac{1}{\sqrt{2 \cdot \pi \cdot 0.37}} \cdot \exp\left[-\frac{\mathbf{t}^2}{2 \cdot (0.37)^2}\right]$$

и **b** – параметр сглаживания, величина которого обычно устанавливается в несколько раз больше величины интервала между точками данных на оси **x** и определяется тем, насколько большое окно желательно использовать при сглаживании.

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 функции **ksmooth(·)**, которая использует фиксированную ширину полосы сглаживания для всех исходных данных, supsmooth(·) адаптивно выбирает различную ширину полосы сглаживания для различных частей исходного вектора данных.

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

Функция ksmooth (vx, vy1, b) возвращает m-мерный вектор, созданный сглаживанием при помощи гауссова ядра данных из vy1, где vx и vy1 – m-мерные векторы вещественных чисел. Параметр b управляет окном сглаживания и должен быть установлен в несколько раз больше величины интервала между точками vx.

Функция supsmooth (vx, vy1) возвращает m-мерный вектор, созданный локальным использованием симметричной линейной процедуры сглаживания методом наименьших квадратов по правилу k-ближайших соседей, в которой k выбирается адаптивно. Векторы vx и vy1 – m-мерные

65

векторы вещественных чисел. Элементы **vx** должны быть расположены в порядке возрастания.



Рис. 6.8. Функция supsmooth (·)

7. Численное решение линейных и нелинейных уравнений и систем

Система MathCAD обладает широкими возможностями для решения как линейных, так и нелинейных уравнений и систем. Можно решать как одно уравнение с одной неизвестной, так и систему уравнений с несколькими неизвестными. Максимальное количество уравнений и неизвестных в одной системе равно пятидесяти.

7.1. Решение уравнения с одной неизвестной

Для решения уравнений с одной неизвестной используются две функции: root(·) и polyroots(·).

Для численного решения любого уравнения вида f(z)=0 применяется функция

```
root(f(z), z).
```

Параметрами этой функции являются:

f(**z**) – скалярная функция от **z**;

z – скалярная переменная.

Так как для нахождения решения применяется итерационный процесс, то перед использованием функции $root(\cdot)$ необходимо задать начальное приближение для z, т.е. число, лежащее в окрестности предполагаемого корня, значение которого присваивается переменной z.

Если уравнение имеет несколько корней, то возвращаемый результат будет зависеть от заданного начального приближения.

Если требуется найти комплексный корень, то начальное приближение нужно задать комплексным. Для нахождения действительного корня начальное приближение может быть как действительным, так и комплексным.

Точность нахождения корня зависит от значения встроенной переменной **TOL**. Допустимые значения **TOL** $\in [10^{-306}, 1]$; чем меньше **TOL**, тем больше времени потребуется для определения корня.

Так как итерационный процесс может быть расходящимся, то в этом случае появится сообщение об ошибке **Not converging** (**Нет сходимости**). Это может произойти в силу следующих причин:

- уравнение не имеет корней;
- точка начального приближения находится слишком далеко от корня (количество итераций в MathCAD ограничено);
- функция f(z) имеет экстремумы или особые точки между начальным приближением и корнями;
- функция f(z) имеет только комплексные корни, а начальное приближение действительное.

Чтобы установить причину ошибки, можно исследовать график функции f(z). Чем точнее задано начальное приближение корня, тем быстрее будет получено точное решение.

Функция **root(·)** является обычной функцией MathCAD, и ее можно включать в определения других функций. Таким образом можно решать уравнения с параметрами.

На рис. 7.1 показан пример определения четырех действительных корней уравнения. Каждая из четырех стрелок начинается в точке начального приближения, а заканчивается в точке соответствующего корня уравнения. Из рисунка видно, что корень, определяемый функцией **root(·)**, не обязательно будет являться ближайшим к точке начального приближения.



Рис. 7.1. Поиск нескольких решений уравнения функцией root(·)

Для нахождения корней алгебраического полинома n-й степени

 $a_n x^n + a_{n-1} x^{n-1} + ... + a_1 x + a_0$

существует функция polyroots(a), параметром которой является векторстолбец коэффициентов полинома a:=(a₀, a₁, ..., a_{n-1}, a_n)^T размерности n+1.

На рис. 7.2. приведены примеры нахождения корней полиномов с действительными и комплексными коэффициентами функцией polyroots(.).

$$2 \cdot x^{7} + 3 \cdot x^{4} - x^{3} + 4 \cdot x^{2} - x - 7 = 0 \qquad x^{3} + (1 + 2 \cdot i) \cdot x^{2} + (-4 + i) \cdot x + 3 \cdot i = 0$$

a1 := $(-7 - 1 - 4 - 1 - 3 - 0 - 2)^{T}$
a2 := $(3i - 4 + i - 1 + 2i - 1)^{T}$
$$= \begin{bmatrix} -1.265 \\ -1 \\ -0.217 - 1.213i \\ 0.85 + 1.048i \\ 0.85 - 1.048i \\ 1 \end{bmatrix}$$

polyroots(a2) = $\begin{bmatrix} -2.451 - 1.249i \\ -0.114 + 0.527i \\ 1.565 - 1.278i \end{bmatrix}$

Рис. 7.2. Нахождение корней полинома функцией polyroots (·)

7.2. Решение систем уравнений

Система MathCAD позволяет решать как системы однородных и неоднородных линейных алгебраических уравнений, так и системы нелинейных уравнений. Максимальное число уравнений и неизвестных в уравнении не должно превышать пятидесяти.

Для решения систем неоднородных линейных алгебраических уравнений вида

 $\begin{cases} a_{1,1}\mathbf{x}_1 + a_{1,2}\mathbf{x}_2 + \dots + a_{1,n}\mathbf{x}_n = \mathbf{b}_1 ; \\ a_{2,1}\mathbf{x}_1 + a_{2,2}\mathbf{x}_2 + \dots + a_{2,n}\mathbf{x}_n = \mathbf{b}_2 ; \\ \dots \\ a_{n,1}\mathbf{x}_1 + a_{n,2}\mathbf{x}_2 + \dots + a_{n,n}\mathbf{x}_n = \mathbf{b}_n ; \end{cases}$

или в матричной форме **Аx** = **b** предназначена функция

lsolve(A,b).

Параметрами этой функции являются:

А – невырожденная матрица размерности $n \times n$, содержащая коэффициенты системы $a_{i,j}$, i=1,2,...,n, j=1,2,...,n;

b – ненулевой вектор-столбец размерности n, содержащий свободные члены b_i, i=1,2,...,n.

Функция **lsolve(.)** возвращает вектор-столбец, содержащий решение системы.

В отличие от функции **lsolve(·)** блок решения уравнений и систем позволяет решать любые системы уравнений с помощью итерационного процесса. Для решения системы с помощью блока решения необходимо:

- задать начальные приближения для всех неизвестных, входящих в систему;
- напечатать ключевое слово Given, указывающее MathCAD на начало блока решения;
- ввести уравнения и неравенства, входящие в систему. При вводе уравнений необходимо использовать полужирный знак равенства, который можно задать с помощью панели операторов отношения или нажатием клавиш [Ctrl]+[=];
- ввести выражение, включающее функцию Find(·).

Заметим, что для ограничивающих неравенств **нельзя** использовать знак ≠ и неравенства вида **a<b<c**. На рис. 7.3. приведен пример решения уравнения с одним неизвестным, а на рис. 7.4. – решение системы уравнений с использованием ограничивающих неравенств.

Решение уравнения $x^2 - 1 = e^{-x} \sin(10 \cdot x)$ x := 1 Given $x^2 - 1 = e^{-x} \sin(10 \cdot x)$ Find(x) = 0.96208

Рис. 7.3. Решение уравнения с одним неизвестным



Рис. 7.4. Использование ограничивающих неравенств

Функция Find(\cdot) должна иметь столько аргументов, сколько неизвестных содержит данная система. Она возвращает решение системы, находящейся в блоке решения. Если система состоит из одного уравнения, то функция Find(\cdot) возвращает скаляр. Если же в систему входит более одного уравнения, то будет возвращен вектор-столбец, размерность которого равна числу неизвестных в системе. Вызов функции Find(\cdot), находящийся после ключевого слова Given, закрывает блок решения. Точность решения задач с помощью блока решения зависит от значения встроенной переменной **TOL**.

Функция **Find(·)**, завершающая **блок решения**, может быть использована аналогично любой другой функции MathCAD, т.е. можно:

вывести найденное решение, напечатав Find (·) =;

- определить переменную с использованием этой функции, завершив блок решения выражением вида a:=Find(·);
- определить другую функцию

f(a1, a2, ..., aM, x1, x2, ..., xN) := Find(x1, x2, ..., xN),

где **a1, a2, ..., аМ** – параметры, передаваемые в **блок решения**; **x1, x2, ..., xN** – неизвестные, относительно которых решается система.

Пример такого определения показан на рис. 7.5.

x := 0 y := 0 Given $x^{2} + y^{2} = a$ x + y = b f(a,b) := Find(x,y) f(4,1) = $\begin{bmatrix} 1.82288 \\ -0.82288 \end{bmatrix}$ f(5,1) = $\begin{bmatrix} 2 \\ -1 \end{bmatrix}$

Рис. 7.5. Определение функции через блок решения

Если в результате решения системы на каком-то шаге не может быть найдено более точное приближение к искомому решению по сравнению с предыдущим шагом, то поиск решения прекращается и MathCAD выдает сообщение об ошибке **Did not find solution** (**Решение не найдено**). Причиной появления такого сообщения об ошибке может быть следующее:

- поставленная задача не имеет решения;
- для системы, не имеющей вещественных корней, в качестве начального приближения хотя бы для одной из неизвестных задано вещественное число;
- в процессе поиска решения последовательность приближений попала в точку локального минимума невязки. Метод поиска решения, используемый в MathCAD, не позволяет в этом случае построить следующее приближение, которое уменьшало бы невязку. Для поиска решения можно попробовать сменить точку начального приближения или добавить ограничения на переменные, чтобы обойти точку локального минимума;
- поставленная задача не может быть решена с точностью, определенной в переменной **TOL**. В этом случае нужно увеличить значение этой переменной.

Функция Minerr(·) также может закрывать блок решения, но, в отличие от функции Find(·), при невозможности дальнейшего уточнения текущего приближения Minerr(·) не выдает сообщения об ошибке Did not find solution (Peшение не найдено), а возвращает это приближение. Minerr(·) обычно возвращает решение, которое минимизирует функционал невязки, но проверка того, является ли возвращаемый ответ абсолютным минимумом для функционала невязки, не выполняется. Поэтому необходимо всегда включать проверку достоверности получаемых результатов. Встроенная переменная ERR дает величину невязки для приближенного решения. Функция Minerr(·) часто используется для решения задач регрессии. Правила использования и параметры функции Minerr(·).

 $x_k := \frac{2 \cdot k}{5}$ z := 0, 0.01..4 $\alpha := 0.4$ $\beta := 2$ k := 0..10 $\mathbf{y} \coloneqq (0.08 \ 0.40 \ 0.55 \ 0.60 \ 0.50 \ 0.25 \ 0.10 \ 0.08 \ 0.02 \ 0.015 \ 0.01 \)^T$ $F(x, \alpha, \beta) := \alpha \cdot \beta \cdot x^{\beta - 1} \cdot e^{-\alpha \cdot x^{\beta}}$ \leftarrow аппроксимирующая функция $K(\alpha, \beta) := \sum_{k} (y_k - F(x_k, \alpha, \beta))^2 \leftarrow$ минимизируемый функционал Given $K(\alpha, \beta) = 0$ 1=1 ← фиктивное уравнение. MathCAD требует, чтобы $a := Minerr(\alpha, \beta)$ число неизвестных совпадало с числом уравнений ERR = 0.01381*←* встроенная переменная, содержащая величину 0.49905 невязки **a** = ← параметры наилучшего приближения 0.8 0.6 $\overset{\mathbf{y}_{k}}{\times \times \times} \\ \mathbf{F}(\mathbf{z}, \mathbf{a}_{0}, \mathbf{a}_{1})$ 0.4 0.2 1 0 2 $\mathbf{X}_{\mathbf{k}}, \mathbf{Z}$

Рис. 7.6. Использование функции Minerr (•) для решения задачи регрессии
На рис. 7.6. приведен пример, в котором функция **Minerr(·)** используется для определения неизвестных параметров в распределении Вейбулла.

8. Решение обыкновенных дифференциальных уравнений (ОДУ) и систем

8.1. Решение задачи Коши

При решении задачи Коши для уравнений первого и высших порядков и систем ОДУ используется функция

rkfixed(y,x1,x2,npoints,D),

которая реализует метод Рунге-Кутта 4-го порядка. Параметры этой функции имеют следующий смысл:

у – **л-мерный вектор** начальных условий (значение решения в точке **х1**);

х1 и **х2** – граничные точки интервала, на котором ищется решение;

npoints – количество точек, в которых ищется решение (не считая **x1**);

D – **имя n**-мерной вектор-функции, имеющей два параметра (**x** и **y**), которая содержит выражения для первых производных неизвестных функций.

Применяемое в описании параметров обозначение **n** – это количество неизвестных функций в уравнении (**n=1**) или системе (**n>1**).

Функция **rkfixed(·)** возвращает матрицу размерности (npoints+1)×(n+1), первый столбец которой содержит значения (npoints+1) точек интервала [x1,x2], в которых определялось решение, а остальные – найденные значения неизвестных функций в этих точках.

8.1.1. Решение ОДУ первого порядка

Задача Коши для обыкновенного дифференциального уравнения первого порядка имеет вид

$$\frac{d}{dx}y(x) = f(x, y), \qquad y(x1) = y_0.$$

Пример решения ОДУ первого порядка функции **rkfixed(·)** приведен на рис. 8.1.



Рис. 8.1. Решение ОДУ первого порядка функцией **rkfixed**(·)

8.1.2. Решение систем ОДУ первого порядка и дифференциальных уравнений высших порядков

Задача Коши для системы **n** обыкновенных дифференциальных уравнений первого порядка имеет вид

$$\frac{d}{dx}y(x) = f(x,y), \qquad y(x1) = y_0,$$

где:

у (х) – вектор-функция, содержащая **n** неизвестных функций скалярного аргумента **x**;

f (x,y) – вектор-функция, содержащая первые производные неизвестных функций y_i(x), i=1,2,...,n;

у₀ – **п**-мерный вектор начальных условий.



Рис. 8.2. Решение системы ОДУ функцией **rkfixed**(·)

Решение систем ОДУ в MathCAD аналогично решению одного дифференциального уравнения. Отличие состоит в том, что у для систем является **n**-мерным вектором, а функция **D** – не скалярной, а **n**-мерной векторфункцией. Пример решения системы ОДУ приведен на рис. 8.2.

Решение ОДУ высших порядков сводится к решению систем ОДУ первого порядка заменой переменных

$$y(x) = y_0(x);$$

$$\frac{d}{dx}y(x) = y_1(x);$$

$$\frac{d^2}{dx^2}y(x) = y_2(x);$$

$$\frac{d^{n-1}}{dx^{n-1}}y(x) = y_{n-1}(x).$$

В п-мерную вектор-функцию D(x, y) заносятся функции, являющие правыми частями уравнений, разрешенных относительно первых производных неизвестных функций $y_0(x), y_1(x), ..., y_{n-1}(x)$. Вектор начальных условий y, состоящий из n компонент, заполняется значениями функций $y_0(x), y_1(x), ..., y_{n-1}(x)$ в начальной точке интервала x1. Функция rkfixed(·) возвращает матрицу размерности (npoints+1)×(n+1), первый столбец которой содержит значения (npoints+1) точек интервала [x1,x2], в которых определялось решение, а остальные n столбцов – найденные значения функций $y_0(x), y_1(x), ..., y_{n-1}(x)$ в этих точках.

Подобной же заменой переменных система ОДУ высшего порядка может быть сведена к системе ОДУ первого порядка.

Пример решения задачи Коши для обыкновенного дифференциального уравнения 4-го порядка приведен на рис. 8.3.



Рис. 8.3. Решение ОДУ 4-го порядка

8.2. Решение краевых задач

Двухточечные краевые задачи могут быть решены с помощью функций **sbval(·)** и **bvalfit(·)**. Каждая из этих функций требует выполнения определенных условий.

Для применения функции **sbval**(·) необходимо, чтобы:

- было задано ОДУ n-го порядка:
- в точке x1 была задана часть информации о значении решения и его первых n-1 производных;
- в точке x2 были известны некоторые значения решения и его первых n-1 производных;
- общее количество заданных условий в точках **x1** и **x2** равнялось **n**.

Если все эти условия выполнены, то для решения краевой задачи можно использовать функцию

sbval(v,x1,x2,D,load,score),

параметры которой имеют следующий смысл:

v – вектор начальных приближений для искомых недостающих начальных значений в точке **x1**;

х1, **х2** – граничные точки интервала поиска решения;

D – имя **n**-мерной вектор-функции **D**(**x**, **y**), содержащей выражения для первых **n** производных неизвестной функции. Параметр **x** является скаляром, а **y** – **n**-мерным вектором (аналогично **rkfixed**(·));

load – имя вектор-функции load (x1, v), возвращающей **n**-мерный **вектор** начальных условий в точке x1. Известные значения начальных условий могут быть либо константами, либо функциями аргумента x1. Неизвестные значения должны быть определены как элементы вектора v (v_0 , v_1 и т.д.), причем индексы элементов должны совпадать с индексами соответствующих значений начальных приближений, записанных в векторе v, который является параметром функции sbval(·);

score – имя вектор-функции **score** (**x**2,**y**), возвращающей вектор, размерность которого равна размерности вектора **v**. Каждый элемент возвращаемого вектора должен содержать разность между заданным значением решения или его производной в точке **x**2 и текущим приближением решения или его производной, т.е. соответствующим элементом вектора **y**. Вектор **score** показывает, насколько значения найденного решения в точке **x**2 близки заданным.

Функция sbval (·) возвращает недостающие начальные условия в точке x1, а не решение ОДУ. Таким образом, она сводит краевую задачу к задаче Коши, что позволяет использовать для решения краевой задачи функцию rkfixed(·).

ORIGIN := 0 Решение дифференциального уравнения у"+5у'+3у = 4 на интервале [0,1]. Краевые условия: y(0) = 0, y'(1) = 1, y''(1) = 2. Заменой переменных уравнение приводится к системе уравнений. Граничные точки интервала : x1 := 0 x2 := 1 Начальные значения для условий, заданных в точке x2: $V := \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ $D(x,y) := \begin{bmatrix} y_1 \\ y_2 \\ 4 - 5 \cdot y_1 - 3 \cdot y_0 \end{bmatrix} \quad \text{load}(x1,V) := \begin{bmatrix} 0 \\ V_0 \\ V_1 \end{bmatrix} \quad \text{score}(x2,y) := \begin{bmatrix} y_1 - 1 \\ y_2 - 2 \end{bmatrix}$ s := sbval(V, x1, x2, D, load, score)Найденные значения для y'(0) и y''(0): s = $\begin{vmatrix} -0.039 \\ -0.949 \end{vmatrix}$ Решение полученной задачи Коши: $y := \begin{vmatrix} s_0 \\ s_1 \end{vmatrix}$ z := rkfixed(y, x1, x2, 20, D) j := 0... 203 $\begin{pmatrix} <3>\\ z \end{pmatrix}$ -1 0.2 0.4 0.6 0.8 1 0 $\left(\begin{array}{c} < 0 > \end{array} \right)$

Рис. 8.4. Решение краевой задачи с помощью функции sbval (·)

Пример поиска недостающих начальных условий функцией **sbval**(·) и последующее решение задачи функцией **rkfixed**(·) приведен на рис. 8.4.

Если для функции **sbval** (·) данных недостаточно, но известны значения решения и его первых **n-1** производных в промежуточной точке **xf** интервала **[x1, x2]**, то можно использовать функцию

bvalfit(v1,v2,x1,x2,xf,D,load1,load2,score).

Эта функция решает краевую задачу методом стрельбы: для точек **x1** и **x2** строится семейство решений, и из него выбирается то решение, которое принимает заданное значение в точке **xf**. Параметры функции **bvalfit(.)**

имеют следующий смысл:

v1 – вектор, содержащий начальные приближения для начальных условий, которые не заданы в точке **x1**;

v2 – вектор, содержащий начальные приближения для начальных условий, которые не заданы в точке **x2**;

х1, **х2** – граничные точки интервала поиска решения;

хf – промежуточная точка на интервале **[x1, x2]**;

D – имя вектор-функции **D**(**x**, **y**), содержащей выражения для первых **n** производных неизвестной функции. Параметр **x** является скаляром, а **y** – **n**-мерным вектором (аналогично **rkfixed**(·));

load1 – имя вектор-функции **load1**(**x1**,**v1**), возвращающей **n**-мерный вектор начальных условий в точке **x1**. Известные значения начальных условий могут быть либо константами, либо функциями аргумента **x1**. Неизвестные значения должны быть определены как элементы вектора **v1** (**v1**₀, **v1**₁ и т.д.), причем индексы элементов должны совпадать с индексами соответствующих значений начальных приближений, записанных в векторе **v1**, который является параметром функции **bvalfit**(·);

load2 – имя вектор-функции **load2(x1,v1)**. Смысл параметров этой функции аналогичен смыслу параметров функции **load1**, но для точки **x2**;

score – имя **n**-мерной вектор-функции **score**(**xf**,**y**). Она определяет, насколько значения решений, проходящих через точки **x1** и **x2**, должны соответствовать решению в точке **xf**. Если необходимо, чтобы решения совпали, то определяется: **score**(**xf**,**y**) := **y**.

Функция **bvalfit(·)** особенно полезна, если производные имеют разрыв в какой-либо промежуточной точке интервала [**x1,x2**]. Пример применения функции **bvalfit(·)** приведен на рис. 8.5.



Рис. 8.5. Применение функции **bvalfit(·)**

8.3. Решение задачи Коши для ОДУ и систем ОДУ со специфическими свойствами

8.3.1. Гладкие решения

Если известно, что решение является гладкой функцией, то лучше использовать функцию **Bulstoer(·)**, реализующую метод Bulirsch-Stoer. В этом случае найденное решение будет более точным, чем то, которое возвратила бы функция **rkfixed(·)**.

Список параметров и вид возвращаемой матрицы у функции **Bulstoer(·)** аналогичны функции **rkfixed(·)**.

8.3.2. Неоднородно изменяющиеся решения

Если известно, что решение на одной части интервала изменяется медленно, а на другой – быстро, то лучше использовать функцию **Rkadapt**(·). Эта функция, в отличие от **rkfixed()**, проводящей вычисления с постоянным шагом, анализирует скорость изменения решения и адаптирует соответственно величину шага сетки. Таким образом, в тех областях, где решение изменяется быстро, функция **Rkadapt(·)** проводит вычисления на более мелкой сетке, чем в тех, где решение изменяется медленно. Это позволяет повысить точность расчетов и уменьшить вычислительные затраты. Необходимо отметить, что сетка является неравномерной только во внутренних расчетах. Решение уравнения будет возвращено В равноотстоящих точках.

Список параметров и вид возвращаемой матрицы у функции **Rkadapt(·)** аналогичны функции **rkfixed(·)**.

8.3.3. Жесткие системы

Если система ОДУ является жесткой, то решение, полученное с помощью функции **rkfixed(·)**, может оказаться неустойчивым. Для решения таких систем специально разработаны две функции: **Stiffb(·)** и **Stiffr(·)**, первая из которых реализует метод Bulirsch-Stoer, а вторая – метод Розенброка.

Вид матрицы, возвращаемой этими функциями, аналогичен виду возвращаемой матрицы функциии **rkfixed(·)**. Первые параметры функций **Stiffb(y,x1,x2,npoints,D,J)** и **Stiffb(y,x1,x2, npoints,D,J)** совпадают по смыслу с соответствующими параметрами функции **rkfixed(·)**. Параметр **J** – это имя матричной функции Якоби **J(x,y)**, которая имеет вид

$$J(x,y) = \begin{pmatrix} \frac{\partial D_1}{\partial x} & \frac{\partial D_1}{\partial y_1} & \frac{\partial D_1}{\partial y_n} \\ \frac{\partial D_2}{\partial x} & \frac{\partial D_2}{\partial y_1} & \frac{\partial D_2}{\partial y_n} \\ \frac{\partial \overline{D_n}}{\partial x} & \frac{\partial \overline{D_n}}{\partial y_1} & \frac{\partial \overline{D_n}}{\partial y_n} \end{pmatrix}$$

8.4. Нахождение приближенного решения в конечной точке

Если требуется определить решение ОДУ или системы только в конечной точке x2, то не имеет смысла применять функции, описанные выше, т.к. они находят значения решения не только в конечной точке, но и в промежуточных точках. Для более эффективного решения этой задачи в системе MathCAD имеются четыре специализированные функции:

```
bulstoer(y,x1,x2,acc,D,kmax,save);
rkadapt(y,x1,x2,acc,D,kmax,save);
stiffb(y,x1,x2,acc,D,J,kmax,save);
stiffr(y,x1,x2,acc,D,J,kmax,save).
```

Каждая из этих функций соответствует одной из функций, описанных в предыдущих разделах, и обладает аналогичными свойствами. Параметры **y**, **x1**, **x2**, **D** и **J** по смыслу совпадают с описанными ранее одноименными параметрами. Параметр **acc** контролирует точность решения, влияя на размер шага. Значения **acc**, близкие к 0.001, приводят к хорошей точности решений. Переменная **kmax** – это максимальное число промежуточных точек, в которых ищется приближенное решение. Значение **kmax** содержит ограничение сверху на число строк матрицы, возвращаемой этими функциями. Величина **save** – это минимально допустимый интервал между точками, в которых ищется приближенное решение. Она определяет нижнюю границу различия между любыми двумя числами в первом столбце возвращаемой матрицы.

9. Решение дифференциальных уравнений в частных производных

В системе MathCAD есть две функции, предназначенные для решения уравнения Пуассона

$$\frac{\partial^2 \mathbf{u}(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}^2} + \frac{\partial^2 \mathbf{u}(\mathbf{x}, \mathbf{y})}{\partial \mathbf{y}^2} = \mathbf{f}(\mathbf{x}, \mathbf{y})$$

и его однородной формы — уравнения Лапласа в квадратной области D, при этом уравнение Пуассона аппроксимируют дискретным уравнением вида

$$a_{j,k} \cdot u_{j+1,k} + b_{j,k} \cdot u_{j-1,k} + c_{j,k} \cdot u_{j,k+1} + d_{j,k} \cdot u_{j,k-1} + e_{j,k} \cdot u_{j,k} = f_{j,k}$$

Функция **relax(.)** предназначена для решения уравнения Пуассона в **квадратной** области **D** при ненулевых краевых условиях методом релаксации. Краевыми условиями в данной задаче являются значения неизвестной функции **u(x,y)** на границе области **D**.



Рис. 9.1. Решение уравнения Пуассона функцией relax (·)

Функция **relax(a,b,c,d,e,f,u,rjac)** имеет следующие параметры:

a,**b**,**c**,**d**,**e** – квадратные матрицы одинаковой размерности, содержащие коэффициенты дискретного аналога уравнения Пуассона;

f – квадратная матрица, содержащая значения правой части уравнения в каждой точке области **D**;

u – квадратная матрица, содержащая граничные условия и начальное приближение для решения внутри области **D**;

rjac – спектральный радиус итераций Якоби. Это число между 0 и 1, которое управляет сходимостью алгоритма релаксации. Оптимальное значение параметра **rjac** зависит от конкретных условий задачи.

Функция **relax**(·) возвращает квадратную матрицу, в которой индексы элемента соответствуют расположению точки в области **D**, а значение элемента – решению задачи в данной точке. Пример использования функции **relax**(·) приведен на рис. 9.1.

Если функция **u**(**x**, **y**) на границе области **D** принимает только **нулевые** значения, то для решения уравнения целесообразно использовать функцию **multigrid**(·), которая зачастую решает задачу быстрее, чем **relax**(·).



Рис. 9.2. Решение уравнения Пуассона функцией multigrid (·)

Функция multigrid (M, ncycle) имеет следующие параметры: М – квадратная матрица размерности n=1+2^k, где k – целое число, содержащая

значения функции **f** (**x**, **y**) в области **D**;

ncycle – число циклов в каждом уровне итераций **multigrid(·)** (рекомендуемое значение равно 2).

Пример использования функции **multigrid(·)** для решения уравнения Пуассона приведен на рис. 9.2.

10. Символьный процессор MathCAD

10.1. Введение в символьную математику

Вычислительное ядро системы MathCAD PLUS 6.0 PRO состоит из числового и символьного процессоров. Под словом процессор здесь понимается программный модуль (не аппаратное обеспечение), предназначенный для обработки некоторого класса входных данных (от англ. to process – обрабатывать).

Результатом работы числового процессора является число или массив чисел (вектор, матрица). При этом для проведения вычислений числовой процессор должен «знать» значения всех используемых в выражении переменных. Одним из недостатков числового процессора является появление вычислительной погрешности, так как различные математические константы (например, иррациональные числа π и **e**) теряют свое точное значение, а абстрактному понятию бесконечности присваивается конкретное конечное числовое значение (в MathCAD $\pm^{\infty} = \pm 10^{307}$). Кроме того, существуют выражения, значения которых числовой процессор не определяет. Например, при вычислении интеграла Пуассона (см. рис. 10.1) происходит переполнение.

В отличие от числового процессора результатом работы символьного процессора является не число, а **выражение**, форма которого может быть задана. Исходное выражение можно упростить, проинтегрировать, разложить в ряд и т.д.

Численное вычисление	Символьное вычисление
\int_{t}^{∞}	$\int_{t}^{\infty} t^{2}$
$e^{\frac{1}{2}} dt =$	$e^{-\frac{t}{2}} dt \rightarrow \sqrt{2} \cdot \sqrt{\pi}$
$\sim -\infty$	$\bullet - \infty$
overflow	



Однако, несмотря на все преимущества символьного процессора, не следует забывать, что принцип его работы сильно отличается от принципа работы числового процессора. Символьный процессор может не найти решения, даже если оно есть.

Существует три способа выполнения символьных преобразований:

- использование символьного знака равенства;
- использование команд опции <u>Symbolic (Символика)</u> главного меню при необходимости высокой степени контроля над символьными преобразованиями;
- объединение числового и символьного процессоров для выполнения одной задачи, причем символьный процессор предварительно упрощает выражение так, чтобы числовому потребовалось меньшее количество вычислений для получения результата.

10.2. Символьные преобразования

10.2.1. Символьный знак равенства

Символьный знак равенства → позволяет запускать символьный процессор для немедленного упрощения и вычисления математического выражения. В отличие от обычного знака равенства, всегда возвращающего число, символьный знак равенства может возвращать выражение.

Чтобы иметь возможность использовать символьный знак равенства, необходимо установить флажок Live Symbolics (Символика) в опции <u>Math</u> (Математика) главного меню. Необходимо заметить, что если в той же опции сброшен флажок <u>Automatic Mode (Автоматический режим</u>), то символьный процессор не начнет производить вычисления немедленно после ввода символьного знака равенства. В данной ситуации необходимо или установить вышеуказанный флажок, или выделить выражение вместе с оператором \rightarrow и нажать клавишу [F9]. Оператор Символьный знак равенства вводится комбинацией клавиш [Ctrl]+[.] или кнопкой \rightarrow на наборной панели операторов отношения. Заметим, что символьный знак равенства применяется только ко всему выражению. Этот знак нельзя применить ни к части выражения, ни к результату предыдущего действия такого же знака. (Обычный знак равенства.)

10.2.2. Ключевые слова символьного процессора

Символьный процессор MathCAD «понимает» 7 ключевых слов, управляющих видом выражения, возвращаемого оператором Символьный знак равенства, и одно ключевое слово, запрещающее символьному процессору оптимизировать последующее выражение. Список всех ключевых слов и их краткие описания приведены в табл. 10.1. Все ключевые слова должны набираться в нижнем регистре, т.е. точно так, как они набраны в таблице. Шрифт при этом значения не имеет. Все ключевые слова должны находиться в математических, а не в текстовых, областях. Ключевые слова действуют только до следующего символа \rightarrow

Таблица 10.1.

Ключевое слово	Краткое описание функции
assume	Указывает рассматривать переменную, следующую после этого слова, как неопределенную, даже если ей предварительно присвоено значение. Кроме этого, используется для наложения ограничений на переменные (области определения функций и т.д.)
complex	Предписывает выполнить символьное преобразование в комплексной плоскости. Результат обычно возвращается в форме a+i ·b
expand	Разлагает все степени и произведения сумм в выражении.
factor	Разлагает на множители выбранное выражение (или на простые множители составное число), если все выражение может быть записано в виде произведения сомножителей
float	Указывает отображать результат в формате числа с плавающей запятой, если это возможно
literally	Запрещает символьному процессору пытаться оптимизировать последующее выражение
series	Разлагает выражение от одной или нескольких переменных в ряд Тейлора в окрестности некоторой точки. По умолчанию разложение является полиномом, степень которого максимальная меньшая шестой
simplify	Упрощает выражение, выполняя арифметические преобразования, сокращая общие множители и используя основные тождества для тригонометрических и обратных функций

Ключевые слова символьного процессора

Когда символьный знак равенства используется для преобразования выражения, MathCAD проверяет, были ли переменные и функции, используемые в выражении, определены в рабочем документе. Если такие определения найдены, то символьный процессор использует их. Все неопределенные переменные и функции участвуют в преобразовании в качестве символов. Есть три исключения из этого правила:

• переменная определена как число, содержащее десятичную точку;

- имя переменной, определенной в документе и используемой в выражении, стоит после ключевого слова assume;
- переменная определена как дискретный аргумент.

Во всех трех случаях переменная используется в выражении как символ. Ключевое слово **assume** может быть также использовано для наложения ограничений на переменные в выражении. Ключевое слово **assume** должно предшествовать любому другому ключевому слову, относящемуся к выражению.

На рис. 10.2 приведены примеры вышеописанных исключений и наложения ограничений на переменную.

x := 4 $y := 3$. $i := 15$	
$(x - 2) \cdot (y - 1) \Rightarrow 2 \cdot y - 2$	MathCAD заменяет переменную х на ее значение, а переменную у не заменяет
assume x	Здесь игнорируется определение х, так как
expand $(x + 2) \cdot (x - 2) \Rightarrow x^2 - 4$	имя переменной следует за словом assume
$2 \cdot \pi \cdot i \cdot x \Rightarrow 8 \cdot \pi \cdot i$	Так как переменная і – дискретный аргумент, то ее определение игнорируется
assume a>0,x	Поскольку a>0 , интеграл сходится.
$\int_{-\infty}^{\infty}$	Определение х игнорируется, так как имя
$x \cdot \begin{vmatrix} e^{a} \cdot dt \Rightarrow \frac{\pi}{a} \end{vmatrix}$	переменной следует за ключевым словом
<i>.</i> 0	assume

Рис. 10.2. Ключевое слово assume

	Применение символьного знака
$(a + b)^2 \Rightarrow (a + b)^2$	равенства к выражению такого типа не
	дает результата
	Использование ключевого слова expand
expand	приводит к тому, что MathCAD
$(a + b)^2 \Rightarrow a^2 + 2 \cdot a \cdot b + b^2$	расписывает выражение в виде
	многочлена
	Ключевое слово factor указывает
factor $a^2 - 2 \cdot a \cdot b + b^2 \Rightarrow (a - b)^2$	MathCAD на необходимость разложения
factor	выражения на множители
$17822078 + (2)(2)^5(7)(12)^2(21)$	Разложение составного числа на простые
$1/8230/8 \rightarrow (2)\cdot(3) \cdot (7)\cdot(13) \cdot (31)$	множители

Рис. 1	10.3.	Ключевые	слова	expand	И	factor
				-		

Ключевые слова **expand** и **factor** применяются, соответственно, для развертывания выражения в многочлен и для разложения выражения на множители.

Примеры использования этих слов приведены на рис. 10.3.

Ключевое слово **float** предписывает, по возможности, отображать последующий результат в формате с плавающей запятой. **Общее количество** цифр в числе указывается, при необходимости, после ключевого слова **float**.

Примеры использования ключевых слов **complex** и **float** приведены на рис. 10.4.

$e^{a + b \cdot i} \Rightarrow exp(a + i \cdot b)$	Применение символьного знака равенства к выражению такого типа не лает результата			
complex factor $e^{a + b \cdot i} \Rightarrow exp(a) \cdot (cos(b) + i \cdot sin(b))$	Ключевое слово complex предписывает MathCAD отображать результат в комплексной форме			
float $x \cdot \sqrt{2} \rightarrow 1.4142135623730950488 \cdot x$	Переменная х не определена, следовательно ключевое слово float на нее не действует			
Получение точного десятичного приближения (до 250 знаков после запятой)				
float 55 $\pi \rightarrow 3.141592653589793238462643383279$	502884197169399375105821			

Рис. 10.4. Ключевые слова complex и float

Необходимо отметить, что вид выражения результата зависит от порядка, в котором перечисляются ключевые слова, применяемые к этому выражению. Например, если во втором примере рис. 10.4 поменять местами ключевые слова complex и factor, то вид выражения-результата изменится.

Ключевое слово **series** используется для разложения выражения в ряд Тейлора в окрестности некоторой точки, максимальная степень которого задается пользователем. По умолчанию MathCAD разлагает выражение в ряд в окрестности точки 0 с максимальной степенью **менее** 6. Примеры разложения выражений в ряд Тейлора приведены на рис.10.5.

В некоторых ситуациях ряд может содержать длинные постоянные коэффициенты в символьной форме. Если так, то полезно использовать после ключевого слова **series** ключевое слово **float** с ограничением на количество значащих цифр в числе. Разложение в окрестности точки x=0 series x $e^{x} \Rightarrow 1 + x + \frac{1}{2} \cdot x^{2} + \frac{1}{6} \cdot x^{3} + \frac{1}{24} \cdot x^{4} + \frac{1}{120} \cdot x^{5}$ Разложение в окрестности точки x=1 series x=1 $log(x) \Rightarrow \frac{(x-1)}{ln(10)} - \frac{1}{2} \cdot \frac{(x-1)^{2}}{ln(10)} + \frac{1}{3} \cdot \frac{(x-1)^{3}}{ln(10)} - \frac{1}{4} \cdot \frac{(x-1)^{4}}{ln(10)} + \frac{1}{5} \cdot \frac{(x-1)^{5}}{ln(10)}$ Разложение с максимальной степенью менее 4 в окрестности точки x=0, y=0 series x,y,4 float 3 $e^{x+y} \Rightarrow 1. + x + y + .5 \cdot x^{2} + y \cdot x + .5 \cdot y^{2} + .167 \cdot x^{3} + .5 \cdot y \cdot x^{2} + .5 \cdot y^{2} \cdot x + .167 \cdot y^{3}$

Рис. 10.5. Ключевое слово series

10.3. Подменю Symbolic главного меню

Оператор Символьный знак равенства весьма удобен для выполнения символьных преобразований, но набор предоставляемых ИМ функций Команды Symbolic главного ограничен. опции (Символика) меню обеспечивают больший значительно контроль над символьными преобразованиями.

В опции **<u>Symbolic</u>** (Символика) содержатся следующие команды, смысл которых достаточно понятен:

Evaluate (Вычислить):

<u>Evaluate Symbolically (Вычислить в символах [Shift]+[F9]);</u>

<u>Complex Evaluation (В комплексном виде);</u>

Floating Point Evaluation... (С плавающей точкой...);

<u>Simplify (Упростить);</u>

Expand Expression (Разложить по степеням);

Factor Expression (Разложить на множители);

Collect on Subexpression (Разложить по подвыражению);

Polynomial Coefficients (Полиномиальные коэффициенты);

Differentiate on Variable (Дифференцировать по переменной);

Integrate on Variable (Интегрировать по переменной);

Solve for Variable (Решить относительно переменной);

Substitute for Variable (Заменить переменную);

Expand to Series... (Разложить в ряд...);

Convert to Partial Fraction (Разложить на элементарные дроби);

Matrix (Матричные операции):

Transpose Matrix (Транспонировать матрицу);

I<u>n</u>vert Matrix (Обратить);

Determinant of Matrix (Определитель);

T<u>r</u>ansforms (Преобразования):

Fourier Transform (Преобразование Фурье);

Inverse Fourier Transform (Обратное преобразование Фурье);

Laplace Transform (Преобразование Лапласа);

Inverse Laplace Transform (Обратное преобразование Лапласа);

<u>Z</u> Transform (Z - преобразование);

Inverse Z Transform (Обратное Z - преобразование);

Derivation <u>F</u>ormat...(Расположение результата);

Derive in Place (Замещать).

Для использования команд из опции **Symbolic (Символика)** главного меню, необходимо:

- ввести выражение;
- выделить объект символьной операции;
- выбрать нужную команду.

После этого MathCAD поместит результат выполнения этой команды в рабочий документ.

Объектом символьной операции может быть переменная, математическое выражение или его часть, функция, результат предшествующей операции и т.д. Если объект не выделен, то нет доступа к командам опции **Symbolic** (Символика) главного меню. Для операций символьного процессора нужно выделить выражение или его часть синей рамкой. Для выделения переменной нужно щелкнуть мышью правее этой переменной для появления вертикальной черты.

Если заданная операция невыполнима, то появится сообщение об ошибке или вернется выделенное выражение без изменения. Последнее означает, что операция задана корректно, но результат не может быть получен.

Имеется важное различие между оператором Символьный знак равенства и командами из опции Symbolic (Символика) главного меню. При

изменении исходного выражения слева от символьного знака равенства, выражение-результат будет немедленно обновлен, если отмечены флажками команды <u>Automatic Mode (Автоматический режим)</u> и Live Symbolics (Символика) опции <u>Math (Математика)</u> главного меню. Выражениерезультат, полученное с использованием команд опции <u>Symbolic (Символика)</u>, автоматически модифицироваться не будет. Более того, при изменении исходного выражения, необходимо вручную удалять старое выражениерезультат из рабочего документа.

10.4. Символьные вычисления

10.4.1. Символьная алгебра

Команда **Evaluate (Вычислить)** работает с любыми математическими выражениями, которые, предварительно, выделяются синей рамкой. Эта команда содержит подменю для конкретизации вида вычислений:

Evaluate Symbolically (Вычислить в символах [Shift]+[F9]) – выполнить символьное вычисление выделенного выражения;

<u>Complex Evaluation (В комплексном виде)</u> – выполнить комплексное преобразование выделенного выражения;

<u>Floating Point Evaluation...</u> (С плавающей точкой...) – выполнить численное вычисление выделенного выражения. В диалоговом окне нужно указать количество цифр, которое должны содержать числовые данные.

Символьный процессор по-разному обрабатывает числа, содержащие десятичную точку и не содержащие ее. Общее правило следующее:

- если символьный процессор получает число, содержащее десятичную точку, то любые возвращаемые численные результаты будут десятичными приближениями точного значения;
- если исходное выражение не содержит чисел с плавающей точкой, то результаты будут всегда, когда возможно, являться точными значениями.

На рис. 10.6 показано влияние десятичной точки на вид выражениярезультата.

$\sqrt{2}$	$\sqrt{2}$	$\sqrt{2}$.	1.4142135623730950488
asin(1)	$\frac{1}{2} \cdot \pi$	asin(1.)	1.5707963267948966192
$\frac{22}{19} + \frac{2}{99}$	$\frac{2216}{1881}$	$\frac{22}{19} + \frac{2.0}{99}$	1.1780967570441254652

Рис.	10.6.	Влияние	десятичной	точки	на	результат
------	-------	---------	------------	-------	----	-----------

Команда <u>Simplify</u> (Упростить) выполняет основные алгебраические и тригонометрические упрощения выбранного выражения, например, использует арифметические преобразования, сокращает общие множители, применяет основные тождества для тригонометрических и обратных функций, упрощает корни и степени. Можно упрощать как все выражение целиком, так и его части. Также могут быть упрощены выражения, содержащие матрицы и векторы.

Обычно упрощенный результат тождествен исходному выражению, но если исходное выражение содержит функции типа квадратного корня или обратных тригонометрических функций, то тождество может оказаться нарушенным. Например, выражение $acos(cos(\alpha))$ упрощается до α , что верно только при $\alpha \in [0; \pi]$.

Команда **Expand Expression (Разложить по степеням)** разлагает все степени и произведения сумм в выражении. Если выражение является дробью, то числитель будет разложен, и выражение будет представлено как сумма дробей. Синусы, косинусы и тангенсы сумм будут разложены, насколько это возможно, в синусы и косинусы одиночных переменных.

Команда <u>Factor Expression (Разложить на множители)</u> разлагает на множители выбранное выражение. Если выражение является целым числом, то оно будет разложено на простые сомножители. Эта команда объединяет сумму дробей в одну дробь и упрощает «многоэтажные» дроби с несколькими дробными чертами.

Весьма полезной является команда <u>Collect on Subexpression (Разложить</u> по подвыражению), позволяющая разложить выражение в полином относительно выделенного подвыражения. Выделенное выражение должно быть либо простой переменной, либо встроенной функцией своего аргумента.

Чтобы преобразовать выражение в сумму элементарных дробей, нужно выделить переменную в знаменателе выражения и выбрать команду Convert to <u>Partial Fraction (Разложить на элементарные дроби)</u>. Символьный процессор MathCAD попробует разложить знаменатель дроби на линейные или квадратичные множители с целыми коэффициентами. Если это удастся, то все выражение будет разложено в сумму дробей с этими множителями в качестве знаменателей. Все коэффициенты исходного выражения должны быть целыми, точнее, не должны содержать десятичных точек, в противном случае будет выдано сообщение об ошибке.

Примеры использования этих команд приведены на рис. 10.7.

Исходное выражен	$\text{He} \left(x^3 + \log(y)^2\right)^3$			
Результат упрощен	ия $\frac{\left(x^{3} \cdot \ln(2)^{2} + 2 \cdot x^{3} \cdot \ln(2) \cdot \ln(5) + x^{3} \cdot \ln(5)^{2} + \ln(y)^{2}\right)^{3}}{\left(\ln(2) + \ln(5)\right)^{6}}$			
Результат разложе по подвыражению (по степеням х)	кения $x^9 + 3 \cdot \frac{\ln(y)^2}{\ln(10)^2} \cdot x^6 + 3 \cdot \frac{\ln(y)^4}{\ln(10)^4} \cdot x^3 + \frac{\ln(y)^6}{\ln(10)^6}$			
Примеры разложен	ий на элементарные дроби			
Исходное выражение	$\frac{x^3 - 5 \cdot x + 2}{x^4 + 2 \cdot x^3 - 7 \cdot x^2 - 20 \cdot x - 12}$			
Результат разложения	$\frac{4}{\left[5 \cdot (x+2)^2\right]} + \frac{59}{(25 \cdot (x+2))} + \frac{7}{(50 \cdot (x-3))} - \frac{3}{(2 \cdot (x+1))}$			
Исходное выражение	$\prod_{k=1}^{3} \frac{2^{k}}{x-k!} \cdot x$			
Результат разложения	$64 + \frac{64}{(5 \cdot (x - 1))} - \frac{128}{(x - 2)} + \frac{3456}{(5 \cdot (x - 6))}$			

Рис. 10.7. Разложения по степеням и на элементарные дроби

Иногда бывает необходимо получить вектор, содержащий коэффициенты полинома. Для выполнения этой операции используется команда **Polynomial Coefficients (Полиномиальные коэффициенты)**. Чтобы получить вектор коэффициентов полинома, нужно выделить переменную, относительно которой записан полином, и выполнить вышеуказанную команду. Результатом будет вектор, содержащий коэффициенты при степенях выделенной переменной, причем элементы вектора будут расположены в порядке возрастания показателей степеней.

Если требуется, чтобы MathCAD рассматривал выражение как полином от некоторой функции, то нужно выделить синей рамкой эту функцию и выполнить команду **Polynomial Coefficients (Полиномиальные коэффициенты)**.

Команда Substitute for Variable (Заменить переменную) заменяет выделенную переменную или имя встроенной функции скалярным подвыражением (или именем), находящимся в буфере обмена. Функцию с аргументами этим подвыражением заменить нельзя. При этом возможна ситуация, когда имя функции будет заменено подвыражением, а не именем другой функции, что не имеет смысла с математической точки зрения. Команда заменяет **все** вхождения переменной или имени функции в исходном выражении.

10.4.2. Выполнение вычислений с использованием встроенных функций и операторов

Как правило, символьный процессор может преобразовать любую комбинацию встроенных в MathCAD функций и операторов. Если результат может быть вычислен точно, то он будет отображен, иначе MathCAD возвратит в качестве ответа исходное выражение.

На наборной панели операторов объектов высшей математики находятся одиннадцать операторов, которые можно разбить на следующие группы:

- операторы суммирования и вычисления произведений;
- дифференцирования;
- интегрирования;
- операторы вычисления пределов.

Вычисление суммы	$\sum_{k=1}^{n} \frac{1}{k} \Rightarrow Psi(n+1) + \gamma$
Вычисление произведения	$\left[\prod_{k=1}^{n} \left(1 - \frac{1}{k+1}\right)\right] \Rightarrow \frac{\Gamma(n+1)}{\Gamma(n+2)}$
Вычисление производной первого порядка	$\frac{\mathrm{d}}{\mathrm{d}x} x^{\mathrm{x}} \to x^{\mathrm{x}} \cdot (\ln(x) + 1)$
Вычисление производной высшего порядка	$\frac{d^{3}}{d x^{3}} \frac{x+3}{x-1} \Rightarrow \frac{6}{(x-1)^{3}} - 6 \cdot \frac{(x+3)}{(x-1)^{4}}$
Вычисление неопределенного интеграла	$\int \ln(x) dx \Rightarrow x \cdot \ln(x) - x$
Вычисление кратного неопределенного интеграла	$\int \int (x+y) dx dy \Rightarrow \frac{1}{2} \cdot x^2 \cdot y + \frac{1}{2} \cdot x \cdot y^2$

Рис. 10.8. Примеры выполнения символьных вычислений

Все вышеуказанные действия могут быть выполнены в символьном виде, при этом вычисление неопределенных интегралов и пределов не поддерживается числовым процессором. Так как данные операторы полностью определяют соответствующие математические действия, то для их символьного выполнения может применяться как символьный знак равенства, так и пункт меню <u>Evaluate Symbolically</u> (Вычислить в символах) команды <u>Evaluate</u> (Вычислить) опции <u>Symbolic</u> (Символика) главного меню.

Примеры вычислений с использованием операторов, находящихся на наборной панели объектов высшей математики, приведены на рис. 10.8 и рис. 10.9. Необходимо отметить, что MathCAD вычисляет неопределенные интегралы, не добавляя к результату неизвестную константу, как того требуют математические правила.

Вычисление определенного интеграла	$\int_{1}^{3} \operatorname{atan}(x) dx \Rightarrow 3 \cdot \operatorname{atan}(3) - \frac{1}{2} \cdot \ln(10) - \frac{1}{4} \cdot \pi + \frac{1}{2} \cdot \ln(2)$
Вычисление кратного определенного интеграла	$\int_0^u \int_0^v x^2 + y^4 dx dy \Rightarrow \frac{1}{5} \cdot v \cdot u^5 + \frac{1}{3} \cdot v^3 \cdot u$
Вычисление предела	$\lim_{\mathbf{X} \to \infty} \left(1 + \frac{1}{\mathbf{x}} \right)^{\mathbf{X}} \to \exp(1)$
Вычисление предела слева	$\lim_{x \to 0^{-}} \frac{1}{x} \to -\infty$
Вычисление предела справа	$\lim_{x \to 0^{+}} \frac{1}{x} \to \infty$

Рис. 10.9. Примеры выполнения символьных вычислений

10.4.3. Вычисление производных и интегралов с помощью команд меню <u>Symbolic</u>

Производные первого порядка и одномерные неопределенные интегралы от заданного выражения могут быть вычислены с помощью команд <u>Differentiate on Variable (Дифференцировать по переменной)</u> и <u>Integrate on</u> Variable (Интегрировать по переменной), соответственно. Для этого необходимо:

- ввести выражение, подлежащее дифференцированию или интегрированию;
- выделить синей рамкой **переменную**, по которой требуется взять производную или вычислить интеграл;

выбрать команду <u>Differentiate on Variable (Дифференцировать по переменной)</u> или <u>Integrate on Variable (Интегрировать по переменной)</u> соответственно.

MathCAD отобразит выражение-результат для производной или первообразной заданного выражения по выделенной переменной в рабочем документе.

В случае, если будет выделено подвыражение, а не переменная, или вообще ничего не будет выделено, вышеуказанные команды опции **Symbolic** (Символика) станут недоступными.

Если исходное выражение содержит более одной переменной, то производная или первообразная будут вычисляться по **выделенной** переменной, а все прочие будут считаться константами.

Всем функциям в исходном выражении, которые не являются встроенными, при взятии производной будет сопоставлен соответствующий оператор дифференцирования.

При интегрировании выражения, которое содержит функции пользователя или неопределенные функции, будет выдано сообщение о том, что первообразная не может быть найдена.

10.5. Символьное решение уравнений, систем уравнений и неравенств

Команда Solve for <u>V</u>ariable (Решить относительно переменной) позволяет решать уравнения или неравенства относительно выделенной переменной и выразить его корни или область истинности через остальные параметры уравнения или неравенства. Кроме того, существует возможность символьного решения системы уравнений с использованием блока решений Given... Find.... Отметим, что MathCAD не решает системы неравенств.

Символьное решение уравнения – задача намного более трудная для компьютера, чем численное решение того же уравнения. Может возникнуть ситуация, когда символьный процессор будет не в состоянии найти корни, в то время как числовой процессор определит их.

10.5.1. Решение уравнения или неравенства относительно переменной

Для решения уравнения или неравенства относительно переменной необходимо:

- ввести левую часть уравнения;
- напечатать один из знаков сравнения =, <, ≤, >, ≥ (используется полужирный знак равенства, который вводится нажатием клавиш [Ctrl]+[=]);

- ввести правую часть уравнения;
- выделить переменную, относительно которой требуется решить уравнение или неравенство;
- выбрать команду Solve for <u>Variable</u> (Решить относительно переменной).

Если уравнение имеет несколько корней, или если область истинности неравенства состоит из нескольких частей, то результат будет представлен в виде вектора. Если правой частью **уравнения** является нуль, то достаточно напечатать левую часть и, выделив переменную, выполнить команду **Solve for** <u>Variable (Peшить относительно переменной)</u>. В этом случае MathCAD будет считать, что выражение **приравнивается** к нулю. Примеры символьного решения уравнения и неравенства приведены на рис. 10.10.

Решение уравнения	$\mathbf{a} \cdot \mathbf{x}^2 + \mathbf{b} \cdot \mathbf{x} + \mathbf{c} = 0$	
	$\left[\frac{1}{(2\cdot a)}\cdot\left(-b+\sqrt{b^2-4\cdot a\cdot c}\right)\right]$	
	$\left[\frac{1}{(2\cdot a)}\cdot\left(-b-\sqrt{b^2-4\cdot a\cdot c}\right)\right]$	
Решение неравенства	$x^{3} - 5 \cdot x^{2} - 2 \cdot x + 24 \le 0$	
	$\begin{bmatrix} x \le -2 \\ (3 \le x) \cdot (x \le 4) \end{bmatrix}$	
В данном случае,	результат решения неравенства означает, ч	IT0

областью истинности неравенства является следующая область:

 $\mathbf{x} \in (-\infty; -2] \cup [3; 4].$

Рис. 10.10. Примеры решения уравнения и неравенства

10.5.2. Символьное решение систем уравнений

Для символьного решения системы уравнений, необходимо:

- напечатать ключевое слово Given в математической области (не в текстовом поле);
- ввести в любом порядке уравнения, входящие в систему (для ввода знака сравнения нужно использовать комбинацию клавиш [Ctrl]+[=]);
- напечатать функцию Find(·), соответствующую системе уравнений (аргументами функции являются переменные, относительно которых решается система);

- нажатием клавиш [Ctrl]+[.] ввести символьный знак равенства (для численного решения системы вводится обычный знак равенства [=]);
- щелкнуть мышью в свободном месте рабочего документа.

MathCAD отобразит в рабочем документе набор решений системы справа от символьного знака равенства. Набор решений системы представляется в виде матрицы, каждый столбец которой является решением системы. Количество элементов в столбце равно количеству аргументов функции **Find(·)**. Если какой-либо из векторов содержит не константы, а выражения от переменной или переменных, то решением будет любой вектор, в котором переменные заменены любыми числами, т.е. система имеет бесконечное множество решений.

Пример символьного решения системы уравнений приведен на рис. 10.11.

Given	$x + a \cdot y = b$	Данная	система	уравнений	решается
	$\mathbf{a} \cdot \mathbf{x} + \mathbf{y} = \mathbf{c}$	относител с – это не	ьно переменн известные ко	ных х и у. Вел эффициенты.	ичины а, b и Результатом
$Find(x,y) \rightarrow$	$\frac{(a + b)}{(-1 + a^2)}$	решения является вектор, содержащий два выражения: одно для х, другое – для у. При подстановке вместо а, b и с любых трех чисел будет получено конкретное решение.			
	$\left\lfloor \frac{(a^{2}b^{2}-c)}{(-1+a^{2})} \right\rfloor$				

Рис. 10.11. Пример решения системы уравнений

10.6. Символьные действия с матрицами

Раскрывающееся подменю команды <u>Matrix</u> (Матричные операции) опции <u>Symbolic</u> (Символика) содержит три команды, предназначенные для работы с матрицами:

T<u>r</u>anspose Matrix (Транспонировать) – символьное транспонирование;

Inverse Matrix (Обратить) – поиск обратной матрицы в символьном виде;

Determinant of Matrix (Определитель) – вычисление определителя матрицы в символьном виде.

Для использования этих команд необходимо выделить синей рамкой всю матрицу и выбрать нужную команду. MathCAD возвратит в рабочий документ матрицу (для первых двух команд) или полином (для третьей команды).

Примеры использования этих команд приведены на рис. 10.12.

Транспонирование: 1 2 x $1 x^2 a$ x^2 1 a транспонированная матрица: 2 1 b 1 a b 1 x a Обращение матрицы: 1 2 3 обратная матрица: $\frac{1}{(-8+4\cdot a+4\cdot x)} \cdot \begin{pmatrix} 1-2\cdot a & 4 & 2\cdot a - 3 \\ -x+3\cdot a & -8 & 3\cdot x - a \\ 2\cdot x - 3 & 4 & 1 - 2\cdot x \end{pmatrix}$ x 1 a 3 2 1 Вычисление определителя: 1 2 x определитель: $-7 - 3 \cdot b + 4 \cdot x \cdot b + 6 \cdot a - x \cdot a$ 4 1 3 a b 1

Рис. 10.12. Примеры символьных преобразований матриц

10.7. Интегральные преобразования

Система MathCAD позволяет находить в символьном виде интегральные преобразования и их обращения трех типов. Команда **T**<u>r</u>ansforms (Преобразования) опции <u>Symbolic (Символика)</u> выводит подменю, которое содержит следующие команды для выбора конкретного преобразования:

Fourier Transform (Преобразование Фурье) – выполнить прямое преобразование Фурье относительно выделенной переменной;

Inverse Fourier Transform (Обратное преобразование Фурье) – выполнить обратное преобразование Фурье относительно выделенной переменной;

Laplace Transform (Преобразование Лапласа) – выполнить прямое преобразование Лапласа относительно выделенной переменной;

Inverse <u>Laplace Transform (Обратное преобразование Лапласа)</u> – выполнить обратное преобразование Лапласа относительно выделенной переменной;

<u>Z</u> Transform (Z - преобразование) – выполнить прямое Z-преобразование относительно выделенной переменной;

Inverse Z Transform (Обратное Z - преобразование) – выполнить обратное Z-преобразование относительно выделенной переменной.

Кроме того, существует шесть функций символьного процессора, которые позволяют выполнять соответствующие преобразования не через

команды **T**<u>r</u>ansforms (Преобразования), а с использованием символьного знака равенства.

10.7.1. Прямое и обратное преобразования Фурье

Для получения прямого преобразования Фурье необходимо:

- ввести исходное выражение;
- выделить синей рамкой переменную, по которой требуется преобразовать выражение;
- выполнить команду <u>Fourier Transform</u> (Преобразование Фурье).

Символьный процессор возвращает функцию от переменной **ω**, определяемую формулой

$$\int_{-\infty}^{\infty} f(t) \cdot e^{-i \cdot \omega \cdot t} dt,$$

где **f(t)** – исходное выражение. Результат преобразования Фурье является функцией переменной ω , так как это имя обычно используется в данном контексте. Если исходное выражение содержит обозначение ω , то символьный процессор возвращает функцию не от ω , а от $\omega\omega$, $\omega\omega\omega$ и т.д.

Для вычисления обратного преобразования Фурье требуется:

- ввести исходный Фурье-образ;
- выделить синей рамкой переменную преобразования;
- выполнить команду Inverse Fourier Transform (Обратное преобразование Фурье).

В качестве результата будет возвращена функция переменной t, задаваемая выражением

$$\frac{1}{2\cdot\pi}\cdot\int_{-\infty}^{\infty}F(\omega)\cdot e^{\mathbf{i}\cdot\omega\cdot\mathbf{t}}\,\mathrm{d}\omega,$$

где **F** (**ω**) – исходный Фурье-образ.

Если исходное выражение содержит обозначение t, то символьный процессор поступает так же, как и в случае с прямым преобразованием.

Функция

fourier(f(t),t, ω)

также предназначена для получения Фурье-образа в символьном виде. Параметрами этой функции являются:

f(t) – функция-оригинал, подлежащая преобразованию;

t – имя переменной, относительно которой задана функция $f(\cdot)$;

ω – имя переменной, относительно которой должен быть определен образ.

Для определения обратного преобразования Фурье используется функция invfourier (F (ω), ω , t),

параметры которой имеют следующий смысл:

F (ω) – функция-образ, подлежащая обратному преобразованию;

ω – имя переменной, относительно которой определен образ;

t – имя переменной, относительно которой должен быть задан оригинал.

Примеры использования функций **fourier(·)** и **invfourier(·)** приведены на рис. 10.13.

Прямое преобразование Фурье:

fourier(
$$\Phi(t-1), t, \omega$$
) $\Rightarrow \pi \cdot \Delta(\omega) - \frac{i}{\omega} \cdot \exp(-i \cdot \omega)$

Обратное преобразование Фурье:

simplify

invfourier
$$\left(\pi \cdot \Delta(\omega) - \frac{i}{\omega} \cdot \exp(-i \cdot \omega), \omega, t\right) \Rightarrow 1 - \Phi(-t+1)$$

Через функции fourier(·) и invfourier(·) можно также определить другие функции:

$$U(\omega) := \text{fourier}(\Phi(t-1) - \Phi(t-2), t, \omega) \Rightarrow i \cdot \frac{(-\exp(-1 \cdot \omega) + \exp(-2 \cdot 1 \cdot \omega))}{\omega}$$
$$U(t) := \text{invfourier}\left(\frac{1}{\omega^2}, \omega, t\right) \Rightarrow \frac{1}{2} \cdot t \cdot (-1 + 2 \cdot \Phi(-t))$$

Рис. 10.13. Примеры использования функций fourier (·) и invfourier (·)

10.7.2. Прямое и обратное преобразования Лапласа

Для получения прямого преобразования Лапласа необходимо:

- ввести исходное выражение;
- выделить синей рамкой переменную, по которой требуется преобразовать выражение;
- выполнить команду <u>Laplace Transform</u> (Преобразование Лапласа).

Символьный процессор возвращает функцию от переменной **s**, определяемую формулой

$$\int_0^\infty f(t) \cdot e^{-s \cdot t} dt,$$

где **f(t)** – исходное выражение. MathCAD возвращает функцию от переменной **s**, так как это имя обычно используется в данном контексте. Если исходное выражение содержит обозначение **s**, то символьный процессор возвращает функцию не от **s**, a от **ss**, **sss** и т.д.

Для вычисления обратного преобразования Лапласа нужно:

- ввести исходное выражение;
- выделить синей рамкой переменную преобразования;
- выполнить команду Inverse Laplace Transform (Обратное преобразование Лапласа).

Результатом является функция переменной t, задаваемая выражением

$$\frac{1}{2 \cdot \pi \cdot i} \cdot \int_{\sigma = i \cdot \infty}^{\sigma + i \cdot \infty} F(s) \cdot e^{s \cdot t} ds,$$

где **F**(s) – исходное выражение; σ – абсцисса абсолютной сходимости функции **F**(s).

Если исходное выражение содержит обозначение t, то символьный процессор поступает так же, как и в случае с прямым преобразованием.

Функции

laplace(f(t),t,s)

И

invlaplace(F(s),s,t)

предназначены для получения прямого и обратного преобразований Лапласа в символьном виде. Параметрами этих функций являются:

f(t) – функция-оригинал;

F(**s**) – функция-образ;

t – имя переменной, относительно которой задана функция $f(\cdot)$;

s – имя переменной, относительно которой определен образ.

Способы использования функций laplace(f(t),t,s) и invlaplace(F(s),s,t) аналогичны приведенным на рис. 10.13.

10.7.3. Прямое и обратное z-преобразования

Для получения прямого z-преобразования (называемого иначе производящей функцией) необходимо:

- ввести исходное выражение;
- выделить синей рамкой переменную, по которой требуется преобразовать выражение;
- выполнить команду <u>Z</u> Transform (Z-преобразование).

Символьный процессор возвращает функцию от переменной **z**, определяемую формулой

$$\sum_{n=0}^{\infty} f(n) \cdot z^{-n} ,$$

где **f**(**n**) – исходное выражение. Результат z-преобразования является функцией переменной z, т.к. это имя обычно используется в данном контексте. Если исходное выражение содержит обозначение z, то символьный процессор возвращает функцию не от z, a от zz, zzz и т.д.

Для вычисления обратного z-преобразования требуется:

- ввести исходный z-образ;
- выделить синей рамкой переменную преобразования;
- выполнить команду <u>Inverse</u> Z Transform (Обратное Zпреобразование).

В качестве результата будет возвращена функция переменной n, задаваемая контурным интегралом в окрестности начала координат

$$\frac{1}{2\cdot\pi\cdot i}\cdot\int_{\mathbf{C}}\mathbf{F}(z)\cdot z^{n-1}\,\mathrm{d}z,$$

где **F**(**z**) – исходный z-образ; **С** – контур, включающий все особые точки подынтегрального выражения.

Если исходное выражение содержит обозначение **n**, то символьный процессор поступает так же, как и в случае с прямым преобразованием.

Функции

```
ztrans(f(n),n,z)
```

И

```
invztrans(F(z),z,n)
```

предназначены для получения прямого и обратного z-преобразований в символьном виде. Параметрами этих функций являются:

f (n) – функция-оригинал;

F (z) – функция-образ;

n – имя переменной, относительно которой задан оригинал;

z – имя переменной, относительно которой определен образ.

Способы использования функций ztrans(f(n), n, z) и invztrans(F(z), z, n) аналогичны приведенным на рис. 10.13.

10.8. Оптимизация математических вычислений

Система MathCAD PLUS 6.0 PRO обладает возможностью объединения числового и символьного процессоров для решения одной задачи. Числовой процессор всегда решает задачи «в лоб», т.е. вычисляет выражение именно в той форме, в которой его ввел пользователь. При проведении относительно простых вычислений это свойство числового процессора особенно не сказывается на количестве времени, которое требуется для решения задачи. Если же необходимо найти значение достаточно сложного выражения численными методами, то затраты времени могут многократно возрасти. Пусть, например, требуется вычислить следующее выражение:

$$\int_{0}^{1} \int_{0}^{1-x} \int_{0}^{1-x-y} \frac{1}{(x+y+z+1)^{3}} dz dy dx$$

Числовой процессор в данном случае решает трудоемкую задачу вычисления численной аппроксимации тройного интеграла. Затраты времени при этом будут достаточно велики, и уменьшить их можно, только уменьшив точность вычисления интеграла, т.е. увеличив значение переменной TOL.

Существует два способа, с помощью которых можно облегчить работу числового процессора:

- использовать режим optimize;
- применить комбинацию знаков := и \rightarrow .

Первый способ заключается в установлении флажка у команды **Optimize** (Оптимизация) опции **Math** (Математика) главного меню. Если этот флажок установлен, то любое выражение, стоящее справа от оператора присваивания := , до обработки числовым процессором будет отправлено символьному процессору для выполнения упрощений. В этом случае справа от исходного выражения появится красная звездочка, указывающая на то, что выражение было оптимизировано символьным процессором. Результат упрощения не

будет отображен в рабочем документе. Для того, чтобы увидеть упрощенный вариант, нужно дважды щелкнуть мышью на красной звездочке. Появится окно



аuto 📕 Рис. 10.14. Символьная оптимизация вычислительной задачи

Live symbolics Popup (см. рис. 10.14.). В этом окне будут находиться комментарии к символьной оптимизации и упрощенный вариант исходного выражения.

Меню окна Live symbolics Popup содержит единственную команду – Сору (Копировать) опции Edit (Редактирование) главного меню, которая позволяет скопировать содержимое окна в буфер обмена. Для того чтобы закрыть это окно, нужно щелкнуть мышью где-нибудь в рабочем документе.

Если используется режим optimize, и при этом требуется вычислить какое-либо выражение без оптимизации, то необходимо перед строкой, содержащей оператор присваивания и исходное выражение поместить ключевое слово literally. Это слово запрещает символьному процессору оптимизировать следующее за ним выражение.

Второй способ оптимизации вычислений заключается в использовании комбинации операторов присваивания и символьного знака равенства.

На рис. 10.15 приведены примеры оптимизации вычислений с использованием второго способа.

При использовании второго способа, может возникнуть проблема со специальными функциями символьного процессора, которые не распознаются числовым процессором. Если выражение-результат оказалось содержащим эти функции, то рекомендуется применить ключевое слово **float**, как показано на рис. 10.15.



Рис. 10.15. Применение второго способа оптимизации

10.9. Использование функций и переменных

10.9.1. Встроенные функции и переменные

Числовой и символьный процессоры MathCAD используют различные множества функций, которые частично пересекаются. Часть встроенных функций, которые распознаются числовым процессором и имеют общепринятое математическое обозначение, например $cos(\cdot)$ или $ln(\cdot)$, распознаются и символьным процессором. Другие функции, такие как, например **polyroots(·)** или **rnorm(·)**, не имеют общепринятого математического обозначения и не рассматриваются символьным процессором. Кроме того, символьный процессор использует ряд *специальных функций символьного процессора*, которые не вычисляет числовой процессор.

Символьный процессор распознает следующие встроенные функции: все тригонометрические, гиперболические и обратные к ним функции; функции **Re(·)** и **Im(·)**; функции **erf(·)**(функция интеграла ошибок), **Г(·)** (гамма-функция Эйлера), **mod(·)**, **Ф(·)** (ступенчатая функция Хэвисайда); **max(·)** и

min(·); матричные функции **identity(·)**, **eigenvals(·)** и **rank(·)**. За исключением трех различий, эти функции трактуются одинаково как числовым, так и символьным процессорами. Эти различия состоят в следующем:

- в отличие от численной функции mod(·) символьная функция mod(·) требует целочисленный второй аргумент, а в качестве первого может принимать выражения;
- некоторые обратные тригонометрические функции используют различные ветви на комплексной плоскости;
- матричная функция rank(·) в символьном виде вычисляет ранг как вещественных, так и комплексных матриц, тогда как числовая функция rank(·) может обрабатывать только вещественные матрицы.

Встроенные константы, имеющие общепринятое математическое значение, трактуются одинаково обоими процессорами. Символьный процессор распознает числа *π*, е и знак ∞. Они сохраняют свой точный смысл при использовании в символьных преобразованиях. Если же константа не является В математике, то она не обрабатывается общепринятой символьным процессором. Например, константы ORIGIN TOL не И являются определенными в символьных преобразованиях.

Функции и переменные, определенные пользователем, распознаются символьным процессором при использовании Символьного знака равенства и не распознаются при использовании команд опции <u>Symbolic (Символика)</u>.

10.9.2. Специальные функции символьного процессора

Результат символьного преобразования иногда оказывается содержащим функции, не используемые числовым процессором. Численные значения таких функций при конкретных значениях аргументов можно присвоить переменным, используя второй способ оптимизации символьных выражений. Если же вычисление значения функции не происходит даже при использовании второго способа, то рекомендуется применить ключевое слово **float** или поставить после какого-либо аргумента десятичную точку.

Перечень всех специальных функций символьного процессора:

• Chi(x) :=
$$\gamma$$
 + ln(x) +
$$\int_0^x \frac{\cosh(t) - 1}{t} dt;$$
•
$$\operatorname{Ci}(x) := \gamma + \ln(x) + \int_0^x \frac{\cos(t) - 1}{t} dt;$$

•
$$\operatorname{csgn}(z) := \operatorname{if}(\operatorname{Re}(z) \neq 0, 2 \cdot \Phi(\operatorname{Re}(z)) - 1, 2 \cdot \Phi(\operatorname{Im}(z)) - 1);$$

• dilog(x) :=
$$\int_{1}^{x} \frac{\ln(t)}{1-t} dt;$$

• Dirac(x) := if(x=0,
$$\infty$$
,0);

• Ei(x) :=
$$\gamma$$
 + ln(x) + $\sum_{n=1}^{\infty} \frac{x^n}{n \cdot n!}$ (x > 0);

•
$$\operatorname{erf}(z) := \frac{2}{\sqrt{\pi}} \cdot \sum_{n=0}^{\infty} \frac{(-1)^n \cdot z^{2 \cdot n + 1}}{n! \cdot (2 \cdot n + 1)}$$
 (z-комплексное);

• FresnelC(x) :=
$$\int_0^x \cos\left(\frac{\pi}{2} \cdot t^2\right) dt;$$

• FresnelS(x) :=
$$\int_{0}^{x} \sin\left(\frac{\pi}{2} \cdot t^{2}\right) dt;$$

• LegendreE(x,k) :=
$$\int_{0}^{x} \sqrt{\frac{1-k^2 \cdot t^2}{1-t^2}} dt;$$

• LegendreEc1 (k) := LegendreEc
$$\left(\sqrt{1-k^2}\right)$$
;

• LegendreF(x,k) :=
$$\int_{0}^{1} \frac{1}{\sqrt{(1-t^2)\cdot(1-k^2\cdot t^2)}} dt;$$

LegendreKc(k) := LegendreF(1,k);

• LegendreKc1(k) := LegendreKc
$$\left(\sqrt{1 - k^2}\right)$$
;

• LegendrePi(x,n,k) := $\int_{0}^{x} \frac{1}{(1-n^2 \cdot t^2) \cdot \sqrt{(1-t^2) \cdot (1-k^2 \cdot t^2)}} dt;$

• LegendrePic1(n,k) := LegendrePic
$$\left(n, \sqrt{1-k^2}\right)$$
;

•
$$Psi(x) := \frac{d}{dx} ln(\Gamma(x));$$

•
$$Psi(n,x) := \frac{d^n}{d x^n} Psi(x);$$

• Shi(x) :=
$$\int_{0}^{x} \frac{\sinh(t)}{t} dt;$$

•
$$\operatorname{Si}(x) := \int_0^x \frac{\sin(t)}{t} dt;$$

- signum(x) := if $\left(x=0, 1, \frac{x}{|x|}\right)$;
- W(x) главная ветвь функции, удовлетворяющей условию W(x)·exp(W(x))=x;

• Zeta(s) :=
$$\sum_{n=1}^{\infty} \frac{1}{n^s}$$
, где s > 1.

Здесь у – константа Эйлера, приблизительно равная 0.5772156649.

10.10. Отображение результатов символьных вычислений

Отображением результатов символьных вычислений управляют две команды опции **Symbolic (Символика**):

- Derivation Format...(Расположение результата...) расположение результата;
- **Derive in Place (Замещать)** замещение исходного выражения.

Выполнение первой команды приводит к появлению на экране диалогового окна **Derivation Format (Расположение результата)**, с помощью которого производится настройка параметров расположения результатов символьных вычислений.

При установке флажка <u>Show derivation comments</u> (Показывать комментарии) в рабочий документ будут вставляться комментарии к символьным преобразованиям, например, yields, expands in partial fractions to, has determinant и т.д.

Кроме того, в диалоговом окне Derivation Format находятся три кнопки в группе Show derivation steps (Располагать результат), которые означают следующее:

- <u>vertically</u>, inserting lines результат отображается ниже исходного выражения, при этом вставляются свободные строки;
- vertically, <u>without inserting lines</u> результат отображается ниже исходного выражения, при этом свободные строки не вставляются;
- <u>h</u>orizontally результат отображается правее исходного выражения.

Если требуется заменить исходное выражение результатом символьного преобразования, то нужно отметить флажком команду Derive in Place (Замещать) диалогового окна Derivation Format (Расположение результата). При этом исходное выражение будет удаляться из рабочего документа в момент выполнения символьного преобразования, и замещаться результатом.

Отметим, что при использовании оператора Символьный знак равенства результат преобразования всегда будет находиться справа от знака \rightarrow .

11. Функции для работы с файлами данных

Система MathCAD обладает возможностью чтения числовых данных из текстового файла в формате ASCII и записи результатов вычислений в такой файл. Функции работы с файлами данных подразделяются на 2 группы:

- функции обработки неструктурированных файлов;
- функции обработки структурированных файлов.

К первой группе относятся функции **READ** (·), **WRITE** (·) и **APPEND** (·), а ко второй – **READPRN** (·), **WRITEPRN** (·) и **APPENDPRN** (·). Функции первой группы читают или записывают одно число за один вызов, функции же второй группы предназначены для чтения или записи матриц за один вызов.

11.1. Аргументы функций работы с файлами

Аргументы всех вышеперечисленных функций называются файловыми переменными. В отличие от других переменных, за файловыми переменными закреплено имя дискового файла, а не числовое значение. По умолчанию файлам данных устанавливается расширение .dat (для функций первой группы) или .prn (для функций второй группы). Расширения файлов данных также могут быть указаны вручную. Для этого необходимо ввести имя файла с расширением вида filename.ext, но при этом в окне MathCAD появится запись filename_{ext} без точки между именем файла и его расширением. Например, функция WRITE(result_{clc}) будет пытаться записать результат вычислений в файл "result.clc".

При работе с документом MathCAD каталогом по умолчанию считается тот, из которого документ был загружен или в который последний раз производилось сохранение документа. Функции доступа к файлам данных работают с файлами из каталога по умолчанию. Для работы с файлом, который расположен в произвольном каталоге, необходимо записать в файловую переменную MathCAD полный путь и имя файла данных. Для этого:

- выберите в меню File пункт ???Associate with variable...??;
- в появившемся диалоговом окне выберите требуемый файл, введите в поле **???MathCAD Variable???** имя файловой переменной MathCAD и нажмите кнопку **???Associate???**.

Теперь для доступа к требуемому файлу данных можно использовать указанную в диалоговом окне файловую переменную.

Необходимо отметить, что число одновременно открытых файлов ограничено как MathCAD, так и операционной системой.

11.2. Неструктурированные файлы

Для чтения данных из неструктурированного файла применяется функция READ(·). При этом:

- каждый вызов функции READ (·) заново открывает файл и начинает читать данные из его начала. Невозможно непосредственно считать два последовательных набора данных из одного файла, используя два раздельных вызова функции READ (·);
- если равенство содержит функцию READ (·) и дискретную переменную, то данные будут считаны из файла последовательно по одному значению для каждого значения дискретной переменной. Если данных меньше, чем значений дискретного аргумента, то MathCAD прекратит считывание, не возвращая при этом никаких сообщений об ошибках.

Например, строка **N**:=**READ**(datafile) читает из файла "datafile.dat" первое число и записывает его в переменную **N**. Последовательность строк:

прочитает из файла "datafile.dat" не более ста значений и запишет их в вектор **Data**, причем значение **Data**₀ будет совпадать со значением переменной **N**.

Запись данных в неструктурированный файл производится функциями **WRITE (·)** и **APPEND (·)**. Отличие между ними состоит в том, что функция **WRITE (·)** перезаписывает файл, а **APPEND (·)** – добавляет данные в файл.

Способ использования этих функций аналогичен способу использования функции **READ** (·), за исключением того, что **WRITE** (·) и **APPEND** (·) должны находиться слева от оператора присваивания, а не справа как **READ** (·). Соответственно, записываемое выражение должно находиться справа. Все числовые данные записываются в неструктурированный файл с максимальной точностью, вне зависимости от глобального формата числа.

11.3. Структурированные файлы

Функции **READPRN (·)**, **WRITEPRN (·)** и **APPENDPRN (·)** предназначены для работы со структурированными файлами данных. Структурированный файл данных – это файл с фиксированным количеством чисел в строке. Например, если экспортировать прямоугольную область из электронной таблицы в текстовый файл, то этот файл будет являться структурированным в смысле MathCAD.

Функция **READPRN (·)** читает весь файл данных, определяет число строк и столбцов в текстовом файле и создает матрицу соответствующей размерности

из этих данных. Все строки текстового файла должны содержать одинаковое количество чисел, иначе будет выдано сообщение об ошибке. Строки, которые не содержат чисел или содержат текст, игнорируются. Равенство, содержащее функцию **READPRN (·)** должно иметь вид, подобный следующему:

M := READPRN(datafile)

Здесь **м** – матрица, создаваемая функцией **READPRN** (·), а **datafile** – имя файла или файловой переменной.

Для записи или добавления данных в структурированный файл используются, соответственно, функции **WRITEPRN** (·) или **APPENDPRN** (·). На формат числовых данных и размер таблиц, записываемых в текстовый файл этими функциями оказывают влияние две встроенные переменные MathCAD: **PRNPRECISION** и **PRNCOLWIDTH**. Первая из этих переменных устанавливает общее количество цифр, которое будут содержать числа в структурированном файле, а вторая – размер колонки чисел в символах, включая пробелы, знак "–" и десятичную точку. Равенство, содержащее функции **WRITEPRN** (·) или **APPENDPRN** (·) должно иметь следующий вид:

WRITEPRN(datafile) := M

APPENDPRN(datafile) := M

Смысл аргументов этих функций тот же, что и для **READPRN (·)**.

Если записываемый массив является составным (т.е. его элементы сами являются массивами) или комплекснозначным, то будет создан ASCII файл специального формата, читать который сможет, скорее всего, только MathCAD.

Необходимо отметить, что при использовании любых функций обработки структурированных файлов не рекомендуется применение дискретных аргументов и нижних индексов. В особенности это относится к функции **WRITEPRN (·)**.

12. Анимация в MathCAD

Система MathCAD содержит встроенную переменную **FRAME**, которая предназначена для управления процессом создания анимационного ролика. Формально, **FRAME** соодержит номер создаваемого в текущий момент кадра. Для создания ролика необходимо выполнить следующие действия:

- создать объект или группу объектов любого типа, вид которых зависит от значения переменной **FRAME**;
- в меню ???Window??? выбрать пункт ???Animation???\???Create??? (появится диалоговое окно создания анимационного ролика);

- с помощью мыши выделить пунктирной рамкой ту часть рабочего документа, которая представляет собой кадр ролика;
- в диалоговом окне установить нижнюю и верхнюю границы изменения переменной FRAME и скорость воспроизведения ролика (в кадрах в секунду);
- нажать кнопку "???Create???".

В диалоговом окне появится уменьшенное изображение кадра, которое будет перерисовываться в зависимости от значения переменной **FRAME** в соответствии с алгоритмом вычисления и отрисовки кадра. Процесс создания занять некоторое время, зависящее от быстродействия ролика может используемого компьютера сложности алгоритма, определяющего И изображение в кадре. В диалоговом окне также отображается текущее значение **FRAME**. Создание ролика будет завершено, когда **FRAME** достигнет своей верхней границы.

Когда ролик будет создан, его можно немедленно воспроизвести или сохранить на диск в формате Windows AVI.

Перед созданием ролика можно выбрать тип программы сжатия видеоизображений, позволяющей уменьшить размер файла AVI. Для этого нужно нажать кнопку "???Options...??" в диалоговом окне создания ролика.

13. Статистические функции

13.1. Статистики совокупностей

Система MathCAD содержит шесть функций для вычисления статистических оценок случайных совокупностей.

mean (A) – возвращает среднее значение элементов массива А размерности **m**×**n** :

$$\operatorname{mean}(\mathbf{A}) = \frac{1}{\mathbf{m} \cdot \mathbf{n}} \cdot \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} \mathbf{A}_{i,j} \cdot \sum_{i=0}^{m-1} \sum_{j=0}^{m-1} \mathbf{A}_{i,j}$$

- median (A) - возвращает медиану элементов массива А размерности mxn. Медианой называется величина, выше и ниже которой вариационном В ряду находится равное количество членов. Если массив А имеет четное число элементов, медиана определяется среднее как арифметическое двух центральных величин.
- var (A) возвращает дисперсию элементов массива A размерности mxn:

$$\operatorname{var}(A) = \frac{1}{m \cdot n} \cdot \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |A_{i,j} - \operatorname{mean}(A)|^2}{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |A_{i,j} - \operatorname{mean}(A)|^2}$$

cvar (A, B) – возвращает ковариацию элементов массивов A и B размерности mxn:

$$\operatorname{evar}(A) = \frac{1}{m \cdot n} \cdot \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left[A_{i,j} - \operatorname{mean}(A) \right] \cdot \left[\overline{B_{i,j} - \operatorname{mean}(B)} \right],$$

где черта указывает на вычисление сопряженной величины.

stdev (A) – возвращает среднеквадратичное отклонение (квадратный корень из дисперсии) элементов массива A размерности mxn:

 $stdev(A) = \sqrt{var(A)}$.

согг (A, B) – возвращает коэффициент корреляции для двух массивов А и В размерности mxn.

13.2. Распределения вероятностей

Система MathCAD содержит несколько функций для работы с плотностями вероятности. Эти функции делятся на три класса:

- плотности распределения вероятности: вероятность того, что случайная величина будет находиться в окрестности определенной точки, пропорциональна плотности распределения вероятности случайной величины в этой точке;
- функции распределения (вероятности): они дают вероятность того, что случайная величина будет принимать значение, меньшее или равное определенной величине. Эти функции получены интегрированием (или суммированием, когда это необходимо) соответствующей плотности вероятности по подходящему интервалу значений;
- обращения функций распределения: они позволяют по заданной вероятности вычислить такое значение, что вероятность того, что случайная величина будет меньше или равна этому значению, будет равна вероятности, заданной в качестве аргумента.

13.2.1. Функции плотности распределения вероятности

dbeta (x, s1, s2) – возвращает плотность вероятности бета-распределения

$$\frac{\Gamma(s1+s2)}{\Gamma(s1)\cdot\Gamma(s2)}\cdot x^{s1-1}\cdot(1-x)^{s2-1},$$

где s1, s2>0 являются параметрами формы, 0<x<1.

dbinom (k, n, p) – возвращает P(X=k), когда случайная величина X имеет биномиальное распределение

$$\frac{n!}{k! \cdot (n-k)!} \cdot p^k \cdot (1-p)^{n-k} ,$$

где n и k являются целыми числами, удовлетворяющими условию 0≤k≤n. Значение р удовлетворяет неравенству 0≤p≤1.

dcauchy (x, l, s) – возвращает плотность вероятности распределения Коши

$$\pi \cdot \mathbf{s} \cdot (1 - \left(\frac{(\mathbf{x} - \mathbf{l})}{\mathbf{s}}\right)^2)^{-1}$$
,

где 1 является параметром расположения, а s>0 параметр масштаба.

dchisq (x, d) – возвращает плотность вероятности для хи-квадрат распределения

$$\frac{e^{\frac{-x}{2}}}{2\cdot\Gamma\cdot\left(\frac{d}{2}\right)}\cdot\left(\frac{x}{2}\right)^{\frac{d}{2}-1},$$

где d > 0 является числом степеней свободы, x > 0.

dexp (x, r) – возвращает плотность вероятности экспоненциального распределения

$$\mathbf{r} \cdot \mathbf{e}^{-\mathbf{r} \cdot \mathbf{x}}$$

где r>0, x>0.

dF (x, d1, d2) – возвращает плотность вероятности F-распределения

$$\frac{d1^{\frac{d1}{2}} \cdot d2^{\frac{d2}{2}} \cdot \Gamma\left(\frac{d1-d2}{2}\right)}{\Gamma\left(\frac{d1}{2}\right) \cdot \Gamma\left(\frac{d2}{2}\right)} \cdot \frac{x^{\frac{d1-2}{2}}}{(d2+d1\cdot x)^{\frac{d1+d2}{2}}}$$

где d1, d2>0 являются числами степеней свободы, x>0. dgamma (x, s) − возвращает плотность вероятности Г-распределения

$$\frac{\mathbf{x}^{\mathbf{s}-\mathbf{l}}\cdot\mathbf{e}^{-\mathbf{l}}}{\Gamma(\mathbf{s})},$$

в котором s>0 является параметром формы и $x\geq 0$.

dgeom (k, p) – возвращает P(X=k), когда случайная величина X подчиняется геометрическому распределению

$$\mathbf{p} \cdot (1-\mathbf{p})^k$$
,

в котором 0<p≤1 является вероятностью успеха в отдельном испытании, k есть неотрицательное целое число.

dlnorm (x, μ, σ) – возвращает плотность вероятности логнормального распределения

$$\frac{1}{\sqrt{2\cdot\pi}\cdot\sigma\cdot x}\cdot\exp\left[-\frac{1}{2\cdot\sigma^2}\cdot(\ln(x)-\mu)^2\cdot\operatorname{right}\right],$$

в котором μ равно натуральному логарифму среднего значения, $\sigma\!\!>\!\!0$ равно натуральному логарифму среднеквадратичного отклонения и $x\!\!>\!\!0$.

dlogis (l, s) – возвращает плотность вероятности логистического распределения

$$\frac{\exp(-(x-1)/s)}{s \cdot (1 + \exp(-(x-1)/s))^2},$$

в котором 1 является параметром расположения и s>0 – параметром масштаба.

dnbinom (k, n, p) – возвращает P(X=k), когда случайная величина X имеет отрицательное биномиальное распределение

$$\left(\frac{n+k-1}{k}\right)\cdot p^n\cdot (1-p)^k,$$

в котором $0 \le p \le 1$, а n и k являются целыми числами, $n \ge 0$ и k \ge 0.

dnorm (x, μ, σ) -возвращает плотность вероятности нормального распределения

$$\frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma \cdot x} \cdot \exp \left[-\frac{1}{2 \cdot \sigma^2} \cdot (x - \mu)^2 \cdot \operatorname{right} \right]$$

в котором μ и σ есть среднее значение и среднеквадратичное отклонение $\sigma > 0$.

dpois (k, λ) – возвращает P(X=k), когда случайная величина X имеет распределение Пуассона

$$\frac{\lambda^k}{k!} \cdot e^{-\lambda}$$

в котором $\lambda > 0$, а k является неотрицательным целым числом.

dt (x, d) – вычисляет плотность вероятности t-распределения Стьюдента

$$\frac{\Gamma((d+1)/2)}{\Gamma(d/2)\cdot\sqrt{\pi\cdot d}}\cdot\left(1+\frac{x^2}{d}\right)^{-(d+1)/2}$$

,

в котором d является числом степеней свободы, d>0, а х вещественное число. dunif (x, a, b) – вычисляет плотность вероятности равномерного распределения

$$\frac{1}{d-a}$$

в котором b и а являются граничными точками интервала, a
d> и a≤x≤b.

dweibull (x, s) – вычисляет плотность вероятности распределения Вейбулла

$$s \cdot x^{s-1} \cdot exp(-x^s)$$
,

в котором s>0 есть параметр формы и x>0.

13.2.2. Функции распределения

- **спогт (x)** возвращает стандартную нормальную функцию распределения.
- **pbeta (x, s1, s2)** возвращает функцию бета-распределения с параметрами формы s1, s2>0.
- **pbinom (k, n, p)** возвращает функцию биномиального распределения для k успехов в n испытаниях, n натуральное число, p вероятность успеха, 0≤p≤1.
- **pcauchy (x, l, s)** возвращает функцию распределения Коши с параметром масштаба s>0 и параметром расположения 1.
- pchisq (x, d) возвращает функцию распределения хи-квадрат, в котором d >0 равно числу степеней свободы.
- **рехр (х,r)** возвращает функцию экспоненциального распределения, в котором r>0 является параметром.
- **pF (x, d1, d2)** возвращает функцию F-распределения, в котором d1,d2>0 являются числами степеней свободы.
- **pgamma (x, s)** возвращает функцию Гамма-распределения, в котором s>0 является параметром формы.
- **pgeom (k, p)** возвращает функцию геометрического распределения, где р есть вероятность успеха в одиночном испытании, 0<p≤1.

- **plnorm (x, μ, σ)** возвращает функцию логнормального распределения, в котором μ равно логарифму среднего значения, а σ>0 есть логарифм среднеквадратичного отклонения.
- **plogis (x, l, s)** возвращает функцию логистического распределения. 1 есть параметр расположения, s>0 параметр масштаба.
- **pnbinom (k, n, p)** возвращает функцию отрицательного биномиального распределения, в котором 0<p≤1, n натуральное число.
- **рпогт** (**x**, μ , σ) возвращает функцию нормального распределения со средним μ . и среднеквадратичным отклонением $\sigma > 0$.
- **ppois** (k, λ) возвращает функцию распределения Пуассона. $\lambda > 0$.
- **pt (x, d)** возвращает функцию t-распределения Стьюдента. d>0 есть число степеней свободы.
- **punif (x, a, b)** возвращает функцию равномерного распределения. b и а есть граничные точки интервала, a<b.
- **pweibull (x, s)** возвращает функцию распределения Вейбулла. s>0.

13.2.3. Обращения функций распределения

- **qbeta (p, s1, s2)** обращает бета-распределение с параметрами формы $s1,s2>0. 0 \le p \le 1.$
- qbinom (p, n, r) возвращает число успехов в n испытаниях схемы Бернулли при условии, что вероятность успехов не превышает р и г вероятность успеха на одиночном испытании. 0≤r≤1 и 0≤p≤1. n натуральное число.
- **qcauchy (p, l, s)** обращает распределение Коши с параметром масштаба s>0 и параметром расположения 1. 0 .
- qchisq (p, n) обращает хи-квадрат распределение, в котором d>0 является числом степеней свободы. 0≤p<1.
- **qexp (p, r)** обращает экспоненциальное распределение, в котором r>0 является параметром. $0 \le p \le 1$.
- **qF (p, d1, d2)** обращает F-распределение, в котором d1,d2>0 являются числами степеней свободы. 0≤p<1.
- **qgamma (p, s)** обращает Гамма-распределение, в котором s >0 является параметром формы. $0 \le p < 1$.

- **qgeom (p, r)** обращает геометрическое распределение. r есть вероятность успеха при одиночном испытании. 0<p<1 и 0<r<1.
- qlnorm (p, μ, σ) обращает логнормальное распределение, в котором μ является натуральным логарифмом среднего значения, σ>0 – натуральный логарифм среднеквадратичного отклонения. 0≤p<1.</p>
- **qlogis (p, l, s)** обращает логистическое распределение. 1 параметр расположения, s >0 параметр масштаба. 0<p<1.
- **qnbinom (p,n,r)** обращает отрицательное биномиальное распределение с числом испытаний п и вероятностью успеха в одиночном испытании r. 0<r<1 и 0≤p≤1.
- **qnorm (p, \mu, \sigma)** обращает нормальное распределение со средним μ и среднеквадратичным отклонением σ . 0<p<1 и σ >0.
- **qpois** (\mathbf{p}, λ) обращает распределение Пуассона $\lambda > 0$ и $0 \le p \le 1$.
- **qt (p, d)** обращает t-распределение Стьюдента. Р число степеней свободы. d>0 и 0< p <1.
- **qunif (p, a, b)** обращает равномерное распределение. b и a граничные точки интервала. a < b и 0 .
- qweibull (p, s) обращает распределение Вейбулла. s>0 и 0<p<1.

13.3. Функция hist

Функция hist(int,A) используется для вычисления частотного распределения, применяемого для построения гистограмм. Она возвращает вектор, представляющий частоты, с которыми величины, содержащиеся в векторе A, попадают в интервалы, представляемые вектором int. Элементы векторов A и int должны быть вещественными и расположенными в порядке возрастания. МаthCAD интерпретирует int как набор точек, определяющих последовательность интервалов в гистограмме.

Возвращаемый результат является вектором, содержащим на один элемент меньше, чем **int**. Каждый элементы этого вектора содержит число значений A, попавших в интервал, границы которого определены в двух рядом стоящих элементах вектора **int**. При этом данные, меньшие первого и большие последнего элементов вектора **int** игнорируются.

13.4. Случайные числа

Система MathCAD имеет ряд функций для генерирования случайных чисел, подчиняющихся различным законам распределения вероятностей.

Каждая из этих функций создает последовательность псевдослучайных чисел, связанную с некоторым задаваемым стартовым значением. Одно и то же стартовое значение производит одинаковые последовательности чисел. Изменение стартового значения приводит к изменению последовательности генерируемых случайных чисел. Чтобы изменить стартовое значение, нужно в диалоговом окне команды Randomize...(Генератор случайных чисел...) опции Math (Математика) главного меню, ввести новое целочисленное значение. Генератор случайных чисел перезапустится нажатием кнопки OK.

Функции генерирования случайных чисел

- **rbeta (m, s1, s2)** возвращает вектор m случайных чисел, имеющих бетараспределение. s1, s2 > 0 параметры формы.
- **rbinom (m, n, p)** возвращает вектор m случайных чисел, имеющих биномиальное распределение. 0 ≤ p ≤ 1 . n натуральное число.
- rcauchy (m, l, s) возвращает вектор m случайных чисел, имеющих распределение Коши. s > 0 параметр масштаба. l параметр расположения.
- rchisq (m, d) возвращает вектор m случайных чисел, имеющих распределение хи-квадрат. d > 0 число степеней свободы.
- **rexp (m, r)** возвращает вектор m случайных чисел, имеющих экспоненциальное распределение. r > 0 параметр распределения.
- **rF (m, d1, d2)** возвращает вектор m случайных чисел, имеющих F-распределение. d1, d2 > 0 числа степеней свободы.
- **rgamma (m, s)** возвращает вектор m случайных чисел, имеющих гаммараспределение. s > 0 параметр формы.
- **rgeom (m, p)** возвращает вектор m случайных чисел, имеющих геометрическое распределение. 0 .
- **rlnorm (m, μ, σ)** возвращает вектор m случайных чисел, имеющих логнормальное распределение, в котором μ является натуральным логарифмом среднего значения, а σ > 0 натуральный логарифм среднеквадратичного отклонения.
- rlogis (m, l, s) возвращает вектор m случайных чисел, имеющих логистическое распределение, в котором l является параметром расположения, а s > 0 параметр масштаба.

- **rnbinom (m, n, p)** возвращает вектор m случайных чисел, имеющих отрицательное биномиальное распределение. 0 . n натуральное число.
- **гпогт (m, \mu, \sigma)** возвращает вектор m случайных чисел, имеющих нормальное распределение. $\sigma > 0$.
- **гроіз (m, \lambda)** возвращает вектор m случайных чисел, имеющих распределение Пуассона. $\lambda > 0$.
- rt (m, d) возвращает вектор m случайных чисел, имеющих t-распределение Стьюдента. d > 0.
- **runif (m, a, b)** возвращает вектор m случайных чисел, имеющих равномерное распределение, в котором b и а являются граничными точками интервала. a < b.
- rnd (x) возвращает равномерно распределенное случайное число между 0 и х . Эквивалентна функции runif(1,0,x).
- rweibull (m, s) возвращает вектор m случайных чисел, имеющих распределение Вейбулла, в котором s > 0 является параметром формы.

14. Справочная система

Интерактивная справочная система MathCAD достаточно детальна и позволяет пользователю, владеющему английским языком, оперативно ознакомиться с ее возможностями.

14.1. Подменю <u>Н</u>еlp главного меню

Основные возможности справочной системы сосредоточены в опции <u>Help</u> (Справка) главного меню. Она имеет следующие команды:

Index...(Индекс... [F1]) – показать все темы справки;

<u>К</u>еуboard...(Клавиатура...) – показать темы по назначению клавиши быстрого управления;

<u>Using Help...(О справке...)</u> – показать инструкции по использованию справки;

QuickSheet...(Шпаргалки...) – открыть собрание примеров применения MathCAD в стандартных ситуациях;

<u>T</u>echnical Support...(Техническая поддержка...) – открыть информацию об аппаратных средствах;

<u>About MathCAD...(О программе...)</u> – показать информационное окно данной версии MathCAD.

Команда **QuickSheets...(Шпаргалки...)** открывает окно с перечнем более чем 140 небольших и наглядных примеров применения системы. Можно включить в их перечень и свои собственные примеры. Просмотром примеров можно управлять с помощью достаточно понятных пиктограмм, расположенных в верхнем левом углу окна.

Как примеры целиком, так и их части можно копировать в свой рабочий документ. Для этого необходимо:

- с помощью мыши выделить области, подлежащие копированию, выделяющим прямоугольником;
- держа кнопку мыши нажатой, переместить указатель мыши из окна QuickSheets в нужное место рабочего документа;
- освободить кнопку мыши.

Большинство тем QuickSheets имеет кнопку, нажатие которой копирует содержимое окна QuickSheets в буфер обмена. Нужно дважды щелкнуть мышью на такой кнопке. Если теперь вывести курсор за пределы окна QuickSheets, он примет вид стрелки, нагруженной книжкой. Это означает, что курсор несет информацию. Переместив его в нужное место рабочего документа, и, дважды щелкнув мышью можно вставить туда переносимую информацию.

Для копирования и переноса информации должен быть открыт только один рабочий документ. При этом необходимо упорядочить окно рабочего документа и окно QuickSheets для того, чтобы они находились вместе на экране. Для этого нужно воспользоваться командами Tile Horizontal (По горизонтали) или Tile Vertical (По вертикали) опции Window (Окна) главного меню.

14.2. Контекстно – чувствительная справка

В MathCAD можно получать контекстно-чувствительные справки. Для команд меню достаточно установить указатель мыши на команде и прочитать строку сообщения в нижней части окна. Для пиктограмм вывода наборных панелей и кнопок на этих панелях, панели инструментов и средств управления шрифтами достаточно установить над ними указатель мыши и появится короткое сообщение. Для того чтобы получить более детальную справку о командах меню, операторах и сообщениях об ошибках, необходимо щелкнуть мышью на сообщении об ошибке, именах встроенной функции, переменной или оператора и нажать клавишу [F1] для появления окна **Help (Справка)** прямо на нужной информации. Для того чтобы получить более детальную справку о команде меню или о любой из кнопок палитры, необходимо нажать клавиши [Shift]+[F1] для изменения вида указателя мыши на знак вопроса. При выборе команды из меню открывается окно с соответствующей информацией, а

если щелкнуть мышью на кнопке палитры, то появляется имя оператора и соответствующая комбинация клавиш в строке сообщений. Для продолжения работы в документе, нужно нажать клавишу [Esc].

15. Электронные книги

Для обучения работе с системой MathCAD существуют электронные книги. Электронными книгами здесь называются взаимосвязанные наборы объединенных в тематические разделы документов, относящихся к различным сферам применения системы. Документы вызываются друг из друга, обеспечивая гипертекстовые перекрестные ссылки. Для работы с электронными книгами служит опция **Books (Книги)** главного меню, которая содержит следующие команды:

Open Book...(Открыть книгу...) – открыть электронную книгу;

History...(Хронология...) – перечислить все разделы книги, которые просматривались с момента ее открытия;

Search <u>Book...(Искать по книге...)</u> – произвести поиск по всем разделам открытой электронной книги;

Annotate <u>Book</u> (Аннотировать) – выполнить аннотирование электронной книги с сохранением;

Annotate Option... (Опции аннотирования...) – установка опций аннотирования электронных книг:

<u>Save Edited Section (Сохранить раздел)</u> – записать отредактированный сеанс работы;

Save All <u>Edits</u> (Coxpaнить все) – записать все редакторские изменения;

View <u>Original Section (Раздел в оригинале)</u> – выполнить обзор оригинального сеанса работы;

<u>View Edited Section (Измененный раздел)</u> – выполнить обзор отредактированного сеанса работы;

<u>Restore Original Section (Восстановить раздел)</u> – восстановить оригинальный сеанс работы;

Restore Original Book (Восстановить книгу) – восстановить книгу в оригинальном виде;

<u>Highlight Edit (Выделить изменения)</u> – выделить цветом изменения в книге;

Desktop Reference – справочник по математическим расчетам;

Books Sampler – книга примеров применений системы;

Tutorial – самоучитель по правилам работы с системой.

Электронные книги MathCAD делают доступным для использования в рабочих документах множество формул, справочных данных и диаграмм. Для этого можно использовать буфер обмена Windows или копировать и переносить информацию из электронной книги аналогично тому, как это описано для окна **QuickSheets...(Шпаргалки...)** опции **Help (Справка)** главного меню.

16. Перечень ошибок и комментарии к ним

Array size mismatch – несовпадение размеров массивов. Попытка произвести операцию с векторами или матрицами, размеры которых не подходят для этой операции.

Cannot be defined – не может быть определено. Слева от символа определения := находится неопределенное выражение. MathCAD допускает слева от символа определения следующие виды выражений:

- простое имя переменной;
- имя переменной с нижним индексом;
- имя переменной с верхним индексом;
- матрицу имён переменных. Матрица может содержать лишь простые имена переменных или имена переменных с нижними индексами;
- имя функции с аргументами.

Cannot take subscript – не может содержать нижних индексов. Нижний индекс использован не для вектора или матрицы, а для чего-то другого.

Cannot take superscript – не может содержать верхних индексов. Верхний индекс использован не для матрицы, а для чего-то другого.

Definition stack overflow – переполнение стека определений. Использовано слишком много вложенных функций.

Did not find solution – решение не найдено. Система или не имеет решения или блок решения системы выдал в качестве решения приближенный результат, превышающий заданную точность. В последнем случае можно воспользоваться функцией **Minerr** вместо **Find** или увеличить значение встроенной переменной TOL.

Dimension to non-real power – размерность в невещественной степени. Выражение с единицами измерений возведено в комплексную степень. Если выражение имеет размерность, оно может быть возведено только в вещественную степень, иначе MathCAD не может определить единицы, в которых выражен результат.

Domain error – ошибка области определения. Попытка вычислить значение функции, имеющей аргумент, выходящий за область определения. Например, попытка вычислить ln(0).

Duplicate – дублирование. Попытка определить одну переменную дважды в одном определении. Это сообщение появляется, когда создается вектор слева от определения и используется одно и то же имя в этом векторе дважды. Например:

 $(A \ A) := (1 \ 2)$ $(A \ B) := (1 \ 2)$ A = 1 B = 2duplicate

End of file – конец файла. Появляется при попытке считать из файла данных больше значений, чем в нём содержится.

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. Все, начинающееся с цифры, воспринимается, как константа. Если ввести цифру и непосредственно за ней несколько букв, то MathCAD возвратит сообщение об ошибке. Исключение: буквы M, L, T, Q и K, являющиеся обозначениями первичных единиц измерения.

Error in list – ошибка в списке. Указанная функция содержит некорректный список аргументов. Это сообщение появляется также, если создан недопустимый список в другом контексте, например неверный список выражений для координатной оси графика.

Error in solve block – ошибка в блоке решения. Можно видеть это сообщение при вычислении пользовательской функции, выражаемой через блок решения уравнений, содержащий ошибку. Для устранения этой ошибки нужно устранить ошибку в блоке решения уравнений. Если использовать блок решения уравнений непосредственно, не определяя через него функцию, можно получить детализированное диагностическое сообщение.

File error – ошибка файла. Система столкнулась с ошибкой при чтении файла с помощью функций **READ** или **READPRN**. Файл должен быть файлом ASCII, содержащим числа, разделённые пробелами или символами табуляции.

File not found – файл не найден. Система не нашла файл данных, указанный в качестве параметра для функций **READ**, **READPRN**, **APPEND** или **APPENDPRN**, 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.

Illegal context – неверный контекст. Оператор или функция использованы в запрещенном контексте. Например, можно видеть это сообщение в следующих случаях:

- Точка с запятой использована где-либо вне корректного определения диапазона. (Точка с запятой в этом случае выводится на экран как многоточие). Можно использовать точку с запятой только в определении диапазона для дискретного аргумента.
- Функции WRITE или APPEND использованы где-либо вне левой стороны определения. Эти функции не могут применяться в выражениях или в правой части определения.
- Имя существующей функции использовано как имя переменной или имя существующей переменной использовано как имя функции.
- В блоке решения уравнений использовано ограничение типа *≠*.

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.

Illegal ORIGIN – неверное употребление ORIGIN. Переменная ORIGIN определёна через нецелое значение или значение с величиной большей, чем 16000000. Это сообщение появляется при первом использовании индекса после неверного употребления ORIGIN.

Illegal range – неправильный диапазон. Дискретный аргумент определён неправильно. При определении диапазона можно использовать одну из следующих форм записи:

Rvar := n1..n2 Rvar := n1,n2..n3

В определении диапазона допустимо использовать максимум одну запятую и одну точку с запятой. Если используется вторая форма записи, величина n2 должна лежать между значениями n1 и n3, но не равняться n1.

Illegal tolerance – некорректная точность аппроксимации. Встроенная переменная TOL определена вне допустимого диапазона.

Incompatible units – несовместимые единицы. Отмечает выражение, в котором складываются, вычитаются, или выполняются иные операции с выражениями, имеющими различную размерность. Например, можно видеть это сообщение об ошибке при попытке:

• сложить или вычесть два выражения, имеющие различную размерность;

- создать вектор, матрицу, или таблицу, в которой не все элементы имеют одинаковую размерность;
- создать чертёж, в котором два выражения, имеющие различную размерность, откладываются по одной оси.

Indeterminate dimension – неопределённая размерность. Выражение с единицами измерения возводится в степень, включающую дискретный аргумент или вектор. Если выражение имеет размерность, его можно возводить только в степень с фиксированным вещественным показателем.

Index out of bounds – индекс вне границ. Это сообщение помечает индекс, ссылающийся на несуществующее значение массива. Можно видеть это сообщение при использовании отрицательного верхнего или нижнего индекса (или индекса меньшего, чем ORIGIN, если ORIGIN \neq 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 порядком, который не является целым числом от 0 до 5 включительно.

Interrupted – прервано. Вы прервали работу MathCAD нажатием клавиши [Esc] при выполнении вычислений. Для пересчета помеченного выражения щёлкните мышью на выражении и нажмите [F9].

List too long – длинный входной список. Введено слишком много элементов в списке, разделенном запятыми. Это возможно при попытке вывести на график больше выражений, чем допускается MathCAD, или при попытке создать таблицу с количеством измерений, превышающим 50.

Misplaced comma – неуместная запятая. Запятая использована там, где ее не должно быть. Можно использовать запятую в одном из следующих случаях:

- для разделения аргументов функций;
- для разделения первых двух элементов диапазона в определении дискретного аргумента;
- для разделения величин, откладываемых на чертеже вдоль одной оси;
- для разделения элементов в таблице ввода;
- для разделения нижних индексов элемента матрицы.

Missing operand – пропущенный операнд. В выражении пропущен один из операндов.

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.

Must be a multidimensional array – должно быть многомерным массивом. Следует использовать матрицу, имеющую более, чем одну строку, либо более, чем один столбец.

Must be array – должно быть массивом. Попытка выполнить операцию, определенную только для массивов, со скаляром. Для построения графика поверхности или карты линий уровня отображаемый массив должен иметь по меньшей мере две строки или два столбца.

Must be dimensionless – должно быть безразмерным. Указанное выражение имеет размерность, хотя ситуация требует, чтобы оно было безразмерным. Размерные величины нельзя использовать:

- в качестве аргументов большинства функций;
- в показателе степени;
- в нижних и верхних индексах.

Must be increasing – должно быть возрастающим. Вектор, элементы которого не расположены в порядке строгого возрастания, использован в качестве аргумента одной из функций lspline, pspline, cspline, interp, linterp и hist. Первый аргумент этих функций должен быть вектором со строго возрастающими элементами. При этом следует помнить о том, что, если ORIGIN есть 0, то MathCAD включает в число элементов вектора элемент с нулевым индексом, и если он не определён явно, его значение полагается равным нулю.

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. Это сообщение отмечает чертёж, в котором одна из границ по оси, использующей логарифмический масштаб, равна нулю или отрицательна.

Must be range – должен быть диапазон. Что-либо, не являющееся дискретным аргументом, использовано в месте, где он требуется, например, в качестве индекса для суммирования.

Must be real – должно быть вещественным. Комплексное выражение использовано там, где разрешено только вещественное.

Must be scalar – должно быть скаляром. Векторное или матричное выражение использовано там, где требуется скаляр.

Must be square – должна быть квадратной. Это сообщение об ошибке отмечает операцию, в качестве аргумента которой требуется квадратная матрица, а не прямоугольная.

Must be vector – должно быть вектором. Это сообщение отмечает скаляр или матрицу в операции, требующей векторный аргумент.

Nested solve block – вложенные блоки решений. Ключевое слово Given использовано дважды в строке без последующего вызова функции Find или Minerr. В MathCAD недопустимо использование вложенных блоков решения уравнений, хотя можно определять функции через блоки решения уравнений и затем использовать их в других блоках решения уравнений.

No matching given – нет соответствующего **Given**. Это сообщение указывает на наличие функции **Find** или **Minerr** без соответствующего ей слова **Given**.

Non-scalar value – не скалярная величина. Вектор или выражение, содержащее дискретный аргумент, используются там, где требуется скаляр.

Not a name – не является именем. Число или другая комбинация символов использованы там, где MathCAD требует имя.

Not converging – отсутствует сходимость. MathCAD не способен вычислить результат интегрирования, дифференцирования, функции root, Find или Minerr с требуемой точностью. Попробуйте установить большее значение TOL или установить другое начальное приближение.

Not enough memory – недостаточно памяти. MathCAD не сумел выполнить операцию над массивом из-за нехватки памяти.

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, конечный или промежуточный результат которого превосходит наибольшее число (10³⁰⁷).

Range not allowed – диапазон недопустим. Попытка использования дискретного аргумента внутри блока решения уравнений. Чтобы решать систему уравнений для многих значений параметров, определите функцию через блок решения уравнений.

Significance lost – потеряны значащие цифры. Это сообщение отмечает попытку вывести функцию от величины, которая лежит за пределами диапазона, где значение функции может быть вычислено точно. Например,

 $sin(10^{100}) =$ significance lost

Singularity – особая точка. Попытка вычислить функцию или выполнить операцию с недопустимым значением. Например, можно видеть это сообщение при делении на ноль или попытке обратить вырожденную матрицу (с нулевым определителем).

Stack overflow – переполнение стека. Вычисление выражения привело к переполнению внутреннего стека MathCAD. Это может быть результатом попытки вычисления слишком сложного выражения или рекурсивного определения функции. Упростите выражение или разделите его на несколько подвыражений.

Subscript too large – слишком большой нижний индекс. Попытка использовать нижний индекс, превышающий пределы, допускаемые MathCAD.

Symbolic operator only – только символьный оператор. Попытка получить численный результат у выражения, которое может быть вычислено только символьным процессором.

Too few arguments – слишком мало аргументов. Указанное выражение содержит функцию со слишком малым количеством аргументов. Для встроенных функций число аргументов фиксировано. Нажмите [Shift]+[F1] и щёлкните мышью на имени функции, чтобы получить информацию о её аргументах. Для функций пользователя число параметров зависит от определения, сделанного в рабочем документе.

Too few constraints – слишком мало ограничений. Это сообщение указывает на **Find** или **Given** с количеством ограничений меньшим, чем число переменных. Добавьте несущественные ограничения или уменьшите число переменных, относительно которых ищется решение.

Too few elements – слишком мало элементов. Это сообщение указывает на преобразование Фурье, кубический сплайн или функцию линейной интерполяции, применяемую для вектора со слишком малым числом элементов. Преобразование Фурье и обратное к нему требуют как минимум четыре элемента вектора.

Too few subscripts – слишком мало нижних индексов. Для матрицы использован один нижний индекс. Для указания на элементы матрицы нужно использовать два нижних индекса, разделенных запятой.

Too large to compute – слишком велико, чтобы вычислить. MathCAD не сумел выполнить операцию над массивом из-за нехватки памяти. Попробуйте уменьшить размер массива или освободить часть памяти, удалив какие-нибудь большие массивы или графические изображения.

Too large to display – слишком велико, чтобы отобразить. Попытка вывести вектор или матрицу большего размера, чем допускает MathCAD.

Too many arguments – слишком много аргументов. Указанное выражение содержит функцию со слишком большим количеством аргументов. Для встроенных функций число аргументов фиксировано. Нажмите [Shift]+[F1] и щёлкните мышью на имени функции, чтобы получить информацию о её аргументах. Для функций пользователя число параметров зависит от определения, сделанного в рабочем документе.

Too many constraints – слишком много ограничений. В блоке решения уравнений используется более пятидесяти ограничений.

Too many files – слишком много файлов. Открыто слишком много файлов с использованием таких функций доступа к файлам, как **WRITEPRN**, **READPRN**, или других функции этого типа. Одновременно таким образом может быть открыто не более 30 файлов.

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.

Undefined – не определено (негативное изображение). Показанное в негативном изображении имя функции или переменной не определено. Это сообщение часто означает, что для определения переменной использован знак равенства (=) вместо (:=). Это сообщение также появляется при некорректном использовании переменной в глобальном определении. Если переменная используется в правой части глобального определения, она должна быть определена глобально выше него. Если используется локально определенная переменная или переменная, глобальное определение которой находится ниже места ее использования, MathCAD отмечает, что переменная не определена.

Сообщение «не определено» часто указывает на то, что где-то выше в рабочем документе содержится ошибка. Если определение некорректно, то ниже в документе любые выражения, зависящие от этого определения, показываются в негативном изображении.

Underflow – слишком малый результат. Абсолютная величина вычисленного выражения меньше, чем наименьшее число, которое MathCAD может представить.

Unmatched parenthesis – дисбаланс скобок. Вы ввели или пытались вычислить выражение, содержащее левую скобку без соответствующей ей правой скобки. Исправьте выражение, удалив левую скобку или поставив в нужном месте правую.

Wrong size vector – неверный размер вектора. Это сообщение указывает на функции fft или ifft, аргумент которых имеет число элементов, отличное от допустимого. Если ORIGIN равен нулю, MathCAD автоматически включает элемент с нулевым индексом как компоненту вектора-аргумента.

Заключение

Система MathCAD предназначена для решения широкого спектра задач. Совмещая текст, графику и формулы, записанные в их привычном виде, документ MathCAD выглядит как страницы учебника или научной статьи. При этом формулы являются «живыми» - стоит внести изменения в любую из них, как MathCAD выведет новые результаты и перерисует графики. Привычный вид математических формул, широкие возможности системы в отображении графической информации, представление текстовых комментариев с использованием любых символов, доступных в Windows, обеспечивают достаточно комфортную работу в системе MathCAD, позволяя проводить расчеты любой сложности и готовить документы высокого качества.

Может создаться впечатление, что в системе MathCAD все просто. Однако эта простота является обманчивой. Как и во всех других математических системах, при работе в MathCAD возникает достаточно много проблем. Часть из этих проблем возникает при работе с некоторыми функциями MathCAD, ввиду того, что эти функции не «прозрачны» т. е. нет алгоритмов, реализованных функциями. При решении задач различными итерационными методами, такие вопросы, как сходимость, устойчивость, задание начального приближения, фактически решаются методом проб и ошибок.

При численных расчетах MathCAD достаточно оригинально относится к возникающим неопределенностям, полагая $0 \cdot x = 0$ и 0/x = 0, даже не вычисляя значение **x**, и всегда считает 0/0=0. При этом ошибки не диагностируются, что может привести к неверным результатам.

К сожалению, список возникающих проблем может быть продолжен. Поэтому основной совет при решении задач любого уровня в системе MathCAD – доверяй, но проверяй. Особенно это относится к плохо обусловленным задачам.

Литература

- 1. MathCAD 6.0 PLUS. Финансовые, инженерные и научные расчеты в среде Windows 95. /Пер. англ. М.: Информационно-издательский дом "Филинъ", 1996. 712 с.
- 2. Дьяконов В. П. Справочник по MathCAD PLUS 6.0 PRO. М.: "СК Пресс", 1997. 336 с.
- 3. Решетникова Г.Н., Арцер П.А., Гриднев А.А., Хлебников А.А. MathCAD PLUS 6.0 PRO: Учебное пособие. Томск: Изд-во ТГУ, 1999. –Ч. 1. 25с.
- 4. Решетникова Г.Н., Арцер П.А., Гриднев А.А., Хлебников А.А. MathCAD PLUS 6.0 PRO: Учебное пособие. Томск: Изд-во ТГУ, 1999. –Ч. 2. 43с.

Содержание

Введение		
1. Ha	чальные сведения о системе MathCAD	5
1.1.	Интерфейс системы MathCAD	5
1.2.	Подменю <u>Math</u> главного меню	8
1.3.	Встроенные переменные	9
1.4.	Установка формата вывода чисел	
1.5.	Алфавит, константы, переменные, массивы, дискретные аргументы	
1.6.	Операторы, функции и выражения	
1.7.	Редактирование выражений	
2. Pa	бота с файлами и документами	
2.1.	Подменю <u>F</u> ile главного меню	
2.2.	Подменю <u>E</u> dit главного меню	
2.3.	Подменю <u>Т</u> ехt главного меню	
2.4.	Подменю <u>W</u> indow главного меню	
2.5.	Открытие рабочего документа	
2.6.	Открытие рабочего документа через Internet	
2.7.	Сохранение рабочего документа	
2.8.	Пересылка рабочих документов по электроннои почте	
2.9.	Экспорт раоочего документа	
2.10.	Импорт и экспорт текста.	
2.11.	Поиск текста в рабочего документе	
2.12. 3 On		
J. OI	сраторы и функции для работы с векторами и матрицами	J1
3.1.	Операторы	
3.2.	Векторные функции	
3.3.	Матричные функции	
3.4.	Функции, возвращающие специальные характеристики матриц	
4. Гр	афические возможности MathCAD	
4.1.	Подменю <u>G</u> raphics главного меню	
4.2.	Графики в декартовых координатах	
4.3.	Графики в полярных координатах	
4.4.	Графики поверхностей	
5. Пр	ограммирование в системе MathCAD	
5.1.	Введение в программирование	
5.2.	Этапы написания программы	
5.3.	Составной оператор if otherwise	
5.4.	Циклы в MathCAD	
5.5.	Рекурсивные функции и подпрограммы	
5.6.	Возврат значений из программы	
6. Фу	инкции обработки наборов данных	
6.1.	Интерполяция и функция предсказания	
6.1	.1. Сплайны	

	6.1.2. Линейное предсказание	59
(6.2. Функции регрессии	60
	6.2.1. Линейная регрессия	60
	6.2.2. Полиномиальная регрессия	61
	6.2.3. Обобщенная регрессия	62
(6.3. Функции сглаживания	64
7.	Численное решение линейных и нелинейных уравнений и систем	66
,	7 1 Решение уравнения с олной неизвестной	66
-	7.2 Решение систем уравнений	69
8.	Решение обыкновенных лифференциальных уравнений (ОЛУ) и систем	
		70
2	8.1. Решение задачи коши	
	8.1.1. Решение ОДУ первого поряока	
	8.1.2. Решение систем ОДУ первого поряока и оифференциальных уравнении выс	2000X 7 4
	поряоков	
6	8.2. Решение краевых задач	//
Ċ	6.5. Гешение задачи коши для ОДу и систем ОДу со специфическими своиствами $2 \frac{1}{2}$	01 01 01
	8.3.1. Глаокие решения	01
	8.3.2. Пеоонорооно изменяющиеся решения	01
ç	8.1. Нахожление приближенного решения в конешной тонке	01 82
(о.ч. пахождение приолиженного решения в консчной точке	
9.	Решение дифференциальных уравнений в частных производных	82
10	Э. Символьный процессор MathCAD	85
	10.1. Введение в символьную математику	85
1	10.2. Символьные преобразования	86
	10.2.1. Символьный знак равенства	86
	10.2.2. Ключевые слова символьного процессора	86
1	10.3. Подменю Symbolic главного меню	90
	10.4. Символьные вычисления	92
	10.4.1. Символьная алгебра	
	10.4.2. Выполнение вычислений с использованием встроенных функций и оператор	ров 95
	10.4.3. Вычисление производных и интегралов с помощью команд меню <u>Symbolic</u>	96
	10.5. Символьное решение уравнений, систем уравнений и неравенств	97
	10.5.1. Решение уравнения или неравенства относительно переменной	97
	10.5.2. Символьное решение систем уравнений	98
-	10.6. Символьные действия с матрицами	99
-	10.7. Интегральные преобразования	100
	10.7.1. Прямое и обратное преобразования Фурье	101
	10.7.2. Прямое и обратное преобразования Лапласа	102
	10.7.3. Прямое и обратное z-преобразования	104
	10.8. Оптимизация математических вычислений	105
		107
	10.9. Использование функций и переменных	107
	10.9. Использование функций и переменных	107
]	 10.9. Использование функций и переменных	107 107 108
-	 10.9. Использование функций и переменных	107 107 108 111
11	 10.9. Использование функций и переменных	107 107 107 108 111 112

11 11	 1.2. Неструктурированные файлы 1.3. Структурированные файлы 	
12.	Анимация в MathCAD	
13.	Статистические функции	
13	3.1. Статистики совокупностей	
13	3.2. Распределения вероятностей	
	13.2.1. Функции плотности распределения вероятности	
	13.2.2. Функции распределения	
	13.2.3. Обращения функций распределения	
13	3.3. Функция hist	
13	3.4. Случайные числа	
14.	Справочная система	
14	4.1. Подменю Help главного меню	
14	4.2. Контекстно – чувствительная справка	
15.	Электронные книги	
16.	Перечень ошибок и комментарии к ним	
Зак	слючение	
Лит	гература	

Решетникова Г.Н., Хлебников А.А., Арцер П.А., Гриднев А.А.

MathCAD PLUS 6.0 PRO. Учебное пособие.

Подп. в печать	Тираж экз.
Заказ №	УОП ТГУ, Томск, ул. Никитина, 4