

## 1. Общая характеристика и обзор систем компьютерной математики

Одной из основных областей применения ПК являются математические и научно-технические расчеты. Сложные вычислительные задачи, возникающие при моделировании технических устройств и процессов, можно разбить на ряд элементарных: вычисление интегралов, решение уравнений, решение дифференциальных уравнений и т. д. Для решения таких задач имеется целый ряд различных математических пакетов, реализующих разнообразные численные методы, способных так же производить аналитические математические преобразования. Наиболее известными сегодня являются следующие пакеты: Mathematica (фирма Wolfram Research), Maple (фирма Waterloo Maple Inc), Matlab (фирма The MathWorks), MathCAD (фирма MathSoft Inc).

Пакет Mathcad популярен, пожалуй, более в инженерной, чем в научной среде. Характерной особенностью пакета является использование привычных стандартных математических обозначений, то есть документ на экране выглядит точно так же как обычный математический расчет. Для использования пакета не требуется изучать какую-либо систему команд, как, например, в случае пакетов Mathematica или Maple. Пакет ориентирован в первую очередь на проведение численных расчетов, но имеет встроенный символический процессор Maple, что позволяет выполнять аналитические преобразования. В отличие от упомянутых выше пакетов, Mathcad является средой визуального программирования, то есть не требует знания специфического набора команд, имеет чрезвычайно удобный математико-ориентированный интерфейс и прекрасные средства научной графики.

Система MathCAD существует в нескольких основных вариантах:

- ◆ *MathCAD Standard* – идеальная система для повседневных технических вычислений. Предназначена для массовой аудитории и широкого использования в учебном процессе;
- ◆ *MathCAD Professional* – промышленный стандарт прикладного использования математики в технических приложениях. Ориентирована на математиков и научных работников, проводящих сложные и трудоемкие расчеты.
- ◆ *MathCAD Professional Academic* – пакет программ для профессионального использования математического аппарата с электронными учебниками и ресурсами.

## 2. Обзор возможностей системы Mathcad\*

MathCAD объединяет в себе простой текстовый редактор, математический интерпретатор и графический процессор. Весь функциональный набор возможностей системы можно классифицировать следующим образом:

1. вычислительные функции;
2. графические функции;

3. программирование;
4. сервисные функции;
5. аналитические вычисления.

Вычислительные возможности системы могут, применяются для решения разнообразных задач из области математики, физики, инженерных расчетов. К основным вычислительным функциям можно отнести следующие:

1. вычисление арифметических выражений с различной точностью;
2. вычисление производных (обычных и частных), интегралов (обычных, многомерных и контурных);
3. вычисление суммы и произведения;
4. выполнение операций с размерными величинами и переменными;
5. решение уравнений и неравенств и их систем;
6. решение дифференциальных уравнений и систем дифференциальных уравнений;
7. обработка матриц, векторов и ранжированных переменных;
8. использование встроенных математических функций;
9. создание пользовательских функций;
10. использование символьных преобразований и вычислений.

Графические возможности системы применяются для визуализации результатов вычислений и включают построение двумерных графиков, поверхностей, карт линий уровня, трехместных гистограмм, точечных графиков и графиков векторных полей.

Система позволяет создавать программы, представляющие собой выражения, состоящие из программных конструкций, подобных конструкциям языков программирования. Программные выражения позволяют успешно решать в системе те задачи, которые невозможно вычислить с помощью имеющихся встроенных функций.

Пакет MathCAD позволяет выполнять аналитические (символьные) преобразования. Символьные операции можно разделить на шесть разделов:

1. символьная алгебра (упрощения, раскрытие скобок, разложение на множители, приведение подобных, ряды и т.д.);
1. символьные действия анализа (производные, интегралы, пределы);
2. символьное решение уравнений (решение уравнений и систем уравнений);
3. символьные действия с матрицами (матричная алгебра, транспонирование, обращение, определитель);
4. способы отображения символьных результатов;
5. символьные преобразования (преобразования Фурье, Лапласа, z-преобразования).

### **Концепция построения документа системы**

Документ системы MathCAD строится из областей, которые делятся на вычислительные, графические, текстовые и обрабатываются соответственно тремя различными процессорами системы: вычислительным (математическим), графическим и текстовым. Области создаются средствами формульного, текстового и графического редакторов пакета. Общение пользователя с системой происходит на промежуточном математически ориентированном языке визуального программирования – входном языке системы. Этот язык максимально приближен к обычному математическому языку.

Подготовка вычислительных блоков облегчается благодаря вводу шаблонов. Для этого служат панели (палитры) с набором шаблонов различных математических символов.

В системе действует, в зависимости от режима работы, несколько типов курсоров:

- ◆ курсор мыши;
- ◆ курсор в виде маленького красного крестика, определяющий начало формирования новой области – курсор ввода;
- ◆ курсор в виде синей вертикальной линии, используемый внутри формульной области или графической области при наборе и редактировании, в дальнейшем – курсор редактирования;
- ◆ курсор в виде тонкой красной вертикальной линии, используемый внутри текстовой области при наборе и редактировании, в дальнейшем – текстовый курсор;
- ◆ курсор в виде рамки синего цвета, заключающей внутри себя часть области или область целиком для выполнения над ней операций по редактированию, в дальнейшем – рамка объекта;
- ◆ курсор в виде большого черного перекрестья, используемый внутри области, выделенной пунктирной рамкой, в дальнейшем – курсор области;

В процессе работы с документом используется ручной и автоматический режим вычислений. В автоматическом режиме вычисляются все видимые на экране области документа. При просмотре документа вновь появившиеся области вычисляются автоматически, что не всегда удобно при выполнении громоздких

расчетов. В ручном режиме вычисления производятся нажатием кнопки со знаком = на панели инструментов или клавиши F9.

### 3. Типы данных. Входной язык системы

Входной язык системы MathCAD – интерпретирующего типа. Документ обрабатывается сверху вниз, а в пределах строки слева направо, как только система распознает объект, автоматически запускается внутренняя подпрограмма, выполняющая необходимые действия. Например, вычисление выражения, вывод таблицы и т.д.

MathCAD прежде всего требует от пользователя корректного описания алгоритма решения математической задачи на входном языке, очень напоминающем общепринятый язык описания математических и научно–технических расчетов. Рассматривая входной язык системы как язык программирования, мы можем выделить в нем типичные понятия и объекты, такие, как идентификаторы, константы, переменные, массивы и другие типы данных, операторы и функции, управляющие структуры и т. д.

Алфавит входного языка системы определяет совокупность символов и слов, которые используются при задании команд. Алфавит системы MathCAD содержит:

- ◆ строчные и прописные буквы латинского и греческого алфавитов;
- ◆ арабские цифры от 0 до 9;
- ◆ системные переменные;
- ◆ операторы;
- ◆ имена встроенных функций;
- ◆ спецзнаки;

К укрупненным элементам языка относятся типы данных, операторы, функции пользователя и управляющие структуры.

Типы данных системы MathCAD.

| Простые                                      |                  | Структурированные |         |       |
|--|------------------|-------------------|---------|-------|
| константы                                    | переменные       | дискретные        | массивы | файлы |
| целые  | пользовательские |                   | матрицы |       |
| вещественные (от $10^{-307}$ до $10^{307}$ ) | стандартные      |                   | вектора |       |
| строковые константы ("пример")               |                  |                   |         |       |

**Константы** – поименованные объекты, хранящие некоторые значения, которые не могут быть изменены.

**Переменные** – поименованные объекты, имеющие некоторое значение, которое может изменяться по ходу выполнения программы. В MathCAD тип переменной определяется значением и предварительно не задается. Переменные могут быть числовыми, строковыми, символьными и т.д. Имена констант, переменных и иных объектов называют **идентификаторами**. В MathCAD содержится небольшая группа особых объектов

значения которых определены сразу после запуска программы. Их правильнее считать **системными переменными**.

Обычные переменные отличаются от системных тем, что они должны быть предварительно определены пользователем, т. е. им необходимо присвоить значение.

К базовым операторам системы относятся:  $:=$  – локальное присваивание,  $=$  – оператор вычисления,  $\equiv$  – глобальное присваивание. **Локальное присваивание** ( $:=$ ) распространяет свое действие на область документа расположенную ниже места присваивания. Пример – значение переменной  $b$  на момент вычисления  $c$  неопределенно.

$a := 2$       $c := a + b$   
 $b := 3$



**Глобальное присваивание** ( $\equiv$ ) не зависит от места присвоения и распространяет свое действие на весь

$a := 2$       $c := a + b$   
 $b \equiv 3$



документ.

Переменные могут быть размерными, т. е. характеризоваться не только своим значением, но и указанием физической величины, значение которой они хранят. Для присваивания значений таким переменным используется знак  $\equiv$ . Проведение расчетов с размерными величинами и переменными особенно удобно при решении различных физических задач.

$$kg \equiv 1000 \cdot g \quad P := 100 \cdot \frac{kg}{m^2} \quad P = 9.807 \cdot 10^5 \cdot m^{-1} \cdot s^{-2}$$

#### 4. Работа с дискретными переменными в MathCad, примеры

**Дискретной** называется переменная, содержащая несколько значений, каждое из которых отличается от предыдущего на величину постоянного шага и имеющая начальное и конечное значение. Эти переменные имеют два способа определения:

1.  $Name := Nbegin .. Nend$  ;

2.  $Name := Nbegin, (Nbegin + Step) .. Nend$ .

где **Name** – имя переменной, **Nbegin** – ее начальное значение, **Nend** – конечное значение, **..**(; на клавиатуре, либо **m..n** – в арифметической палитре ) – символ, указывающий на изменение переменной в заданных пределах (вводится клавишей **;**). **Step** – шаг изменения переменной (он должен быть положительным, если **Nbegin < Nend**, или отрицательным в обратном случае).

Если шаг не указывается (1 способ), то он выбирается равным единице. Если **Nbegin < Nend**, то шаг переменной будет равен +1, иначе –1.

Дискретные аргументы значительно расширяют возможности MathCAD, позволяя выполнять многократные вычисления или циклы с повторяющимися вычислениями, формировать векторы и матрицы. Параметрами могут быть константы, переменные, выражения.

Функция дискретного аргумента имеет множество значений, каждое из которых соответствует

$$i := 1.5 .. 6.6 \quad x := 0, \frac{\pi}{5} .. \pi \quad x_n := 1 \quad x_k := 12 \quad n := 5 \quad m := x_n, x_n + \frac{x_k - x_n}{n - 1} .. x_k$$

$i$

|     |
|-----|
| 1.5 |
| 2.5 |
| 3.5 |
| 4.5 |
| 5.5 |
| 6.5 |

$x$

|         |
|---------|
| 0       |
| 0.62832 |
| 1.25664 |
| 1.88496 |
| 2.51327 |
| 3.14159 |

$m$

|      |
|------|
| 1    |
| 3.75 |
| 6.5  |
| 9.25 |
| 12   |

соответствующему значению дискретного аргумента. Дискретная переменная может являться аргументом функции пользователя или индексом для организации одномерных и двумерных массивов.


$$\alpha := 0, \frac{2 \cdot \pi}{10} \dots \pi \quad R(\alpha) := \cos(\alpha) + 1 \quad X(\alpha) := R(\alpha) \cdot \cos(\alpha)$$

| $\alpha$ | $R(\alpha)$ | $X(\alpha)$ |
|----------|-------------|-------------|
| 0        | 2           | 2           |
| 0.62832  | 1.80902     | 1.46353     |
| 1.25664  | 1.30902     | 0.40451     |
| 1.88496  | 0.69098     | -0.21353    |
| 2.51327  | 0.19098     | -0.15451    |
| 3.14159  | 0           | 0           |

## 5. Обработка векторов и матриц в MathCad, примеры

**Массив** – имеющая уникальное имя совокупность конечного числа числовых или символьных элементов, упорядоченных некоторым образом и имеющих определенные адреса. В пакете MathCAD используются массивы двух наиболее распространенных типов: одномерные (векторы), двумерные (матрицы).

Порядковый номер элемента, который является его адресом, называется **индексом**. Индексы могут иметь только целочисленные значения. Они могут начинаться с нуля или другого целого числа, в соответствии со значением системной переменной **ORIGIN**. Значение этой переменной может быть переопределено непосредственно в документе, либо с помощью меню **Math–Options**.

1. векторы и матрицы можно задавать различными способами: с помощью команды **Insert – Matrix**, комбинации клавиш **Ctrl + M**, щелчком на кнопке  панели **Matrix**. В появившемся окне задают размерность массива (**Columns** – столбцы, **Rows** – строки).
2. как переменные с индексами (номер элемента в массиве). Нижний индекс вводится нажатием клавиши [, либо щелчком на кнопке **X<sub>n</sub>** панели **Arithmetic**. Незаданные элементы по умолчанию задаются нулевыми.;
3. как переменная с индексом, а элементы в массив заносятся перечислением, через запятую;
4. с использованием дискретного аргумента, когда имеется некоторая явная зависимость для вычисления

$ORIGIN := 1 \quad i := 1..5$

$k_1 := 12 \quad k_2 := 3.2 \quad k_3 := 9 \quad k_4 := 0.6$

$$k = \begin{bmatrix} 12 \\ 0 \\ 3.2 \\ 9 \\ 0.6 \end{bmatrix} \quad a_i := \begin{bmatrix} 2.5 \\ 56 \\ 56 \\ 2.98 \\ 10 \end{bmatrix} \quad a = \begin{bmatrix} 2.5 \\ 56 \\ 56 \\ 2.98 \\ 10 \end{bmatrix}$$

элементов массива через их индексы.

## 3. Обращение к элементам массива

Массивы могут использоваться в выражениях целиком или поэлементно. Для обращения к элементам массивов нужно указать числовые значения индексов элементов. Также можно обращаться к конкретной строке или столбцу матрицы с помощью верхнего индекса (выделения столбца **CTRL+^**) или нижних индексов.

#### 4. Параллельные вычисления.

$$\begin{aligned}
 a &:= \begin{bmatrix} 1 & 7 & 9 \\ 0 & 6 & 2 \\ 8 & 4 & 3 \end{bmatrix} & a_{0,0} = 1 & a_{2,1} = 4 & (a^{<1>})_1 = 6 & i := 0..2 & f_i := \cos(a_{i,1}) & \text{ORIGIN} := 1 & i := 1..5 & j := 1..4 \\
 v &:= a^{<1>} & v = \begin{bmatrix} 7 \\ 6 \\ 4 \end{bmatrix} & a_{i,0} = \begin{bmatrix} 1 \\ 0 \\ 8 \end{bmatrix} & f = \begin{bmatrix} 0.7539 \\ 0.96017 \\ -0.65364 \end{bmatrix} & a_i := i-2.5 & b_{i,j} := i+j & d_i := \frac{\cos(i-j)}{\deg} \\
 & & & & & a = \begin{bmatrix} 2.5 \\ 5 \\ 7.5 \\ 10 \\ 12.5 \end{bmatrix} & b = \begin{bmatrix} 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 7 \\ 5 & 6 & 7 & 8 \\ 6 & 7 & 8 & 9 \end{bmatrix} & d = \begin{bmatrix} -37.451 \\ -8.337 \\ 48.349 \\ -54.87 \\ 23.381 \end{bmatrix}
 \end{aligned}$$

Большинство вычислений с дискретными переменными и массивами строятся на принципе параллельных (поэлементных) вычислений.

#### 6. Стандартные и пользовательские функции в MathCad, примеры

Стандартные:

1. математические (арифметические, тригонометрические, гиперболические, комплексные, специальные);
  2. векторные и матричные (создания, объединения, работы со строками и столбцами, матричная алгебра, специальные функции);
  3. аппроксимация и интерполяция (интерполяция, аппроксимация, регрессия, сплайны, линейное предсказание);
  4. решение линейных, нелинейных, дифференциальных уравнений и систем; (**odesolve(y,x,n)** n – правая граница интервала; x – переменная относительно которой ищется решение; y – вектор искомой функции; Результат: ДУ относительно X)
- (**rkfixed(y,x1,x2,p,D)** y- вектор начальных условий; x1,x2 – левая и правая границы интервала; p – количество точек внутри интервала на котором ищется решение; D – вектор который содержит первую производную)

Пользовательские:  $A(x) := x^2 - 3x + 2$

#### 7. Создание программных фрагментов в MathCad, примеры

Система Mathcad позволяет задавать функции пользователя используя встроенный язык программирования, что позволяет значительно расширить область применения пакета. Перед тем как использовать программу-функцию нужно ее задать – выполнить описание.

Порядок описания программы-функции Mathcad:

1. ввести имя программы-функции и список формальных параметров.

**имя\_программы (формальные параметры: =);**

2. в панели **Programming**, выбрать **Add line**. На экране появится вертикальная черта и вертикальный столбец с двумя полями ввода для ввода операторов, образующих тело программы-функции;
3. в поле 1 (щелкнув на нем мышью или нажав клавишу [Tab]) ввести первый оператор тела программы-функции. Для вставки дополнительных полей ввода нажать на кнопке Add line;
4. в последнем поле (поле 2), определить возвращаемое через имя программы-функции значение ( см. рис.).

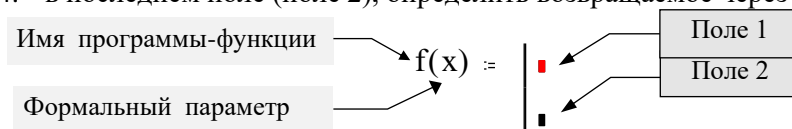


Рис. Структура программы-функции

$$\text{rad\_to\_deg}(x) := \left| z \leftarrow x \cdot \frac{180}{\pi} \right| z$$

$$\text{rad\_to\_deg}\left(\frac{\pi}{4}\right) = 45$$

В качестве примера определим функцию перевода из радиан в градусы:

Для выполнения программы-функции необходимо обратиться к её имени с указанием списка фактических параметров. Фактические параметры указывают при каких конкретных значениях осуществляются вычисления в теле программы. Между фактическими и формальными параметрами должно быть соответствие по количеству, порядку следования и типу. Программный модуль может вести себя и как функция без имени и параметров, но обязательно возвращающий результат.

Переменные определенные в функции являются локальными. Как видно из последнего примера локальное изменение переменной  $x$  в программе–функции, не привело к изменению внешней переменной  $x$ .

## Основные программные операторы

$$\begin{array}{l} s \leftarrow 20 \\ d \leftarrow 15 \\ \sqrt{s^2 + d^2} \end{array} = 25 \quad \begin{array}{l} h := \left| \begin{array}{l} d \leftarrow 1 \\ e^d \end{array} \right| \\ h = 2.718 \end{array} \quad \begin{array}{l} x := 2 \\ f(d, w) := \left| \begin{array}{l} x \leftarrow x + d \\ z \leftarrow \sqrt{x + w} \\ z \end{array} \right| \\ f(10, 5) = 8.464 \quad x = 2 \end{array}$$

Для создания программных модулей используются следующие основные элементы:

| Название  | Описание  |
|-----------|---|
| Add Line  | Создание и расширение программного модуля;  |
| ←         | Внутреннее локальное присваивание.  |
| if        | Оператор условного выражения.   |
| for       | Циклов с заданным числом повторений.  |
| while     | Цикл с предусловием, дейст. пока усл. истинно. Вид: <b>while условие</b> .                                |
| otherwise | Оператор "иначе", обычно используется с <b>if</b> для выполнения действий в случаи не выполнения условия. |
| break     | Вызывает прерывание работы программного блока.  |
| continue  | Используется для продолжения работы цикла после прерывания.   |
| return    | Прерывает выполнение программы и возвращает значение своего операнда.                                     |
| on error  | Оператор  |

## 8.Создание двумерных графиков в MathCad, графики кусочно-непрерывных функций

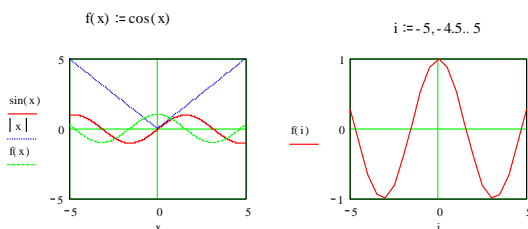
Пакет MathCAD позволяет строить самые разнообразные графики – в декартовой и полярной системе координат, трехмерные поверхности, графики уровней и т. д. **Графические области** делятся на три основных типа – двумерные графики, трехмерные графики и импортированные графические образы. Для построения графиков используются шаблоны. Их перечень содержится в меню **Insert – Graph**:

1. **X-Y Plot[@]** – декартова система координат;
2. **Polar Plot [Ctrl+7]** – график в полярной системе координат;
3. **Surface Plot [Ctrl+2]** – трехмерный график (график поверхности);
4. **Contour Plot [Ctrl+5]** – контурный график трехмерной поверхности (карта линий уровня);
5. **3D Scatter Plot** – график в виде точек в трехмерном пространстве (точечный график фигур);
6. **3D Bar Chart** – гистограмма в трехмерном пространстве;
7. **Vector Field Plot** – график векторного поля на плоскости (векторное поле).

### Построение графиков в декартовой системе координат

В шаблоне графика по вертикали задается через запятую функции, а по горизонтали –переменные. График строится по точкам соединяющихся между собой разнообразными линиями (сплошной, пунктирной и т. д.). Исходные (узловые) точки могут быть показаны в виде жирных точек, квадратиков, кружков и т. д. Крайние шаблоны данных служат для указания предельных значений абсцисс и ординат, т. е. они задают масштабы графика. Если оставить эти шаблоны незаполненными, то масштабы по осям графика будут устанавливаться автоматически.

1. упрощенный способ построения без предварительного задания дискретной переменной (изменение переменной по умолчанию от –10 до +10);



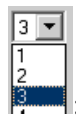
2. с заданием дискретной переменной.

Последовательность действий для построения графика кусочно-непрерывной функции такова:

- с помощью программного фрагмента задать кусочно-непрерывную функцию.

$$y(x) := \begin{cases} 3 \sin(x) & \text{if } x > 25 \\ \frac{x^2}{15} & \text{if } x < 12 \\ (x - 20) & \text{otherwise} \end{cases};$$

- определить аргумент в виде дискретной переменной таким образом, чтобы функция вычислялась по каждой из формул (см. краткие теоретические сведения темы 2), например  $x := 1,1.5 \dots 35$ ;
- построить график заданной функции (см. краткие теоретические сведения темы 5);
- двойным щелчком мыши по области графика открыть окно форматирования;
- для нанесения вспомогательных линий щелчком мыши установить флажок переключателя **Crid Lines** для каждой оси;
- для ввода названий осей и самого графика в окне форматирования выбрать закладку **Labels**, ввести заголовок графика в поле **Title** и установить флажок переключателя **Show Title**;
- ввести названия осей в полях ввода панели **Axis Labels**;
- для изменения толщины линии выбрать закладку **Traces** и задать толщину линии графика, используя список поля **Weight** :



закрывать окно форматирования графика нажатием кнопки **ОК**.

## 9. Редактирование и форматирование графиков в MathCad

Диалоговое окно установки формата отображения графика вызывается из меню **Format – Graph**, либо двойным нажатием левой клавиши мыши на графике.

1. **X-Y Axes** (X-Y Оси) — управление опциями осей;
2. **Traces** (Графики) — управление линиями графика;
3. **Labels** (Надписи) — управление метками (надписями) у осей;
4. **Defaults** (По умолчанию) — задание опций по умолчанию.

### Форматирование осей графика

В панели **X-Y Axes** содержатся следующие основные опции, относящиеся к осям X и Y (**Axis X** и **Axis Y**):

**Log Scale** (Лог. масштаб) — установка логарифмического масштаба;

1. **Crid Lines** (Линии сетки) — установка линий масштабной сетки;
2. **Numbered** (Пронумеровать) — установка цифровых данных по осям;
3. **Autoscale** (Автомасштаб) — автоматическое масштабирование графика;
4. **Show Markers** (Нанести риски) — установка маркеров по осям;
5. **Auto Grid** (Автосетка) — автоматическая установка масштабных линий;
6. **Number of Grids** (Число интервалов) — установка заданного числа масштабных линий.

### Форматирование линий графиков

Следующая панель, **Traces** (Графики), служит для управления отображением линий, которыми строится график. С помощью опций этой панели можно управлять следующими параметрами линий графика:

1. **Legend Label** (Имя кривой) — указание типа линий у оси ординат;
2. **Symbol** (Маркер) — установка символа отметки базовых точек графика;
3. **Line** (Линия) — установка типа линий;
4. **Color** (Цвет) — установка цвета линии и базовых точек;
5. **Type** (Тип) — тип графиков;
6. **Weight** (Толщина) — толщина линий.

без отметки, наклонный крестик, прямой крестик, квадрат, ромбик, окружность.

Графики отдельных функций можно также выделять, используя для их построения линии различного типа: **none** — линия не строится, **solid** — непрерывная линия, **dash** — пунктирная линия, **dadot** — штрих-пунктирная линия

Опция **Color** задает цвета линии и точек красный, синий, зеленый, голубой, коричневый, черный.

Опция **Type** (Тип) задает следующие типы графика:



1. **line** — построение линиями;
2. **points** — построение точками;
3. **err** — построение вертикальными черточками с оценкой интервала погрешностей;
4. **bar** — построение в виде столбцов гистограмм;
5. **step** — построение ступенчатой линией step;
6. **draw** — построение протяжкой от точки до точки.

Возможные конфликты с отметкой символами и типами линий автоматически устраняются. При этом приоритет отдается опции **Type** (Тип), а конфликтные типы линий или точек отмечаются тремя звездочками. Еще две опции связаны с возможностью удаления с графика вспомогательных надписей:

1. **Hide Argument** — прячет обозначения математических выражений по осям графика;
2. **Hide Legend** — прячет обозначения имен кривых графика.

### Задание надписей в графиках

Панель меток **Label** (Надписи). Эта панель появляется, если уже создан текущий график.

Для установки надписей служат небольшие окошки:

1. **Title** - установка титульной надписи к рисунку;
2. **X-Axis** - установка надписи по оси X;
3. **Y-Axis** - установка надписи по оси Y.

## 10 Обработка внешних файлов в Mathcad

| Функция                          | Значение  |
|----------------------------------|---|
| <b>READ</b> ( <i>file</i> )      | Считывает значение из файла данных. Возвращает скаляр. Обычно используется следующим образом: $v_i := \text{READ}(\textit{file})$   |
| <b>WRITE</b> ( <i>file</i> )     | Записывает значение в файл данных. Если файл уже существует, заменяет его на новый файл. Должна использоваться в определениях следующего вида: $\text{WRITE}(\textit{file}) := v_i$   |
| <b>APPEND</b> ( <i>file</i> )    | Дописывает значение к существующему файлу. Должна использоваться в определениях следующего вида: $\text{APPEND}(\textit{file}) := v_i$  |
| <b>READPRN</b> ( <i>file</i> )   | Читает структурированный файл данных. Возвращает матрицу. Каждая строка в файле данных становится строкой в матрице. Число элементов в каждой строке должно быть одинаковым. Обычно используется следующим образом: $\mathbf{A} := \text{READPRN}(\textit{file})$                                   |
| <b>WRITEPRN</b> ( <i>file</i> )  | Записывает матрицу в файл данных. Каждая строка матрицы становится строкой в файле. Должна использоваться в определениях следующего вида: $\text{WRITEPRN}(\textit{file}) := \mathbf{A}$  |
| <b>APPENDPRN</b> ( <i>file</i> ) | Дописывает матрицу к существующему файлу. Каждая строка в матрице становится новой строкой в файле данных. Должна использоваться в определениях следующего вида: $\text{APPENDPRN}(\textit{file}) := \mathbf{A}$ . Существующий файл должен иметь столько же столбцов, как и матрица $\mathbf{A}$ . |

## 11 Информационные технологии: основные понятия и определения

ИТ — совокупность методов и средств использующихся для сбора, хранения, обработки, распространения информации.

ИТ присущи следующие свойства:

- высокая степень расчленённости процесса на стадии, что открывает новые возможности для его рационализации и перевода на выполнение с помощью машин. Это — важнейшая характеристика механизированного технологического процесса;
- системная полнота (целостность) процесса, который должен включать весь набор элементов, обеспечивающих необходимую завершенность действий человека при достижении поставленной цели;
- регулярность процесса и однозначность его фаз, позволяющие применять средние величины при их характеристике, и, следовательно, допускающие их стандартизацию и унификацию. В результате появляется возможность учета, планирования, диспетчеризации информационных процессов.

Информационная система — совокупность средств, методов и персонала, использующих для хранения, обработки и выдачи информации.

*Информационные процессы* — процессы сбора, обработки, накопления, хранения, поиска и распространения информации.

*Документированная информация* – информация, зафиксированная на материальном носителе и имеющая реквизиты для ее идентификации.

*Информационные ресурсы* – отдельные документы и отдельные массивы документов, как сами по себе, так и в информационных системах (библиотеках, архивах, фондах, банках данных, других информационных системах).

## 12. Интегрированная информационная система предприятия

## 13 CAD/CAE и CAM - системы

Под CAD-системами (computer-aided design – компьютерная поддержка проектирования) понимают программное обеспечение, которое автоматизирует труд инженера-конструктора и позволяет решать задачи проектирования изделий и оформления технической документации при помощи персонального компьютера.

CAM-системы (computer-aided manufacturing – компьютерная поддержка изготовления) автоматизируют расчеты траекторий перемещения инструмента для обработки на станках с ЧПУ и обеспечивают выдачу управляющих программ с помощью компьютера.

CAE-системы (computer-aided engineering – компьютерная поддержка инженерных расчетов) предназначены для решения различных инженерных задач, например для расчетов конструктивной прочности, анализа тепловых процессов, расчетов гидравлических систем и механизмов.

## 14 Постановка задачи и обзор численных методов решения ОДУ и систем ОДУ

ДУ – уравнение связывающее между собой независимую переменную  $X$ , искомую функцию  $Y$  и ее производные или дифференциалы.

Решением ДУ называют функцию, которая обращает это уравнение в тождественное.

Общим решением ДУ называют решение, в которое входит столько независимых произвольных постоянных, каков порядок уравнения.

Частным решением ДУ называют решение, полученное из общего при различных числовых значениях произвольных постоянных.

Методы решения:

- метод Эйлера
- модифицированный метод Эйлера
- метод Рунге-Кутты

## 15 Метод Рунге-Кутты для решения ОДУ и систем ОДУ

**rkfixed** – функция для решения ОДУ и систем ОДУ методами Рунге-Кутты четвертого порядка с постоянным шагом. Результат: матрица из  $p+1$  строк, первый столбец – точки в которых получается решение, остальные столбцы – сами решения.

**(rkfixed(y,x1,x2,p,D))**  $y$  – вектор начальных условий;  $x1, x2$  – левая и правая границы интервала;  $p$  – количество точек внутри интервала на котором ищется решение;  $D$  – вектор который содержит первую производную)

**Rkdafter** – с переменным шагом

## 16 Стандартные функции для решения ОДУ в Mathcad

**Odesolve** – блочным методом

**(odesolve(y,x,n))**  $n$  – правая граница интервала;  $x$  – переменная относительно которой ищется решение;  $y$  – вектор искомой функции; Результат: ДУ относительно  $X$ )

**rkfixed** – функция для решения ОДУ и систем ОДУ методами Рунге-Кутты четвертого порядка с постоянным шагом. Результат: матрица из  $p+1$  строк, первый столбец – точки в которых получается решение, остальные столбцы – сами решения.

(rkfixed(y,x1,x2,p,D) у- вектор начальных условий; x1,x2 – левая и правая границы интервала; p – количество точек внутри интервала на котором ищется решение; D – вектор который содержит первую производную)

Rkadafttr – с переменным шагом

## 17 Алгоритм решения ОДУ первого порядка и систем ОДУ в Mathcad. Примеры

1. Given (начало решающего блока)
2. Задаем вид ДУ, а знак равенства с помощью логических операций.
3. Задать начальное условие, так же с помощью логического равно.
4. Задать функцию odesolve с параметрами, формирующие результат функции

Given

$$y'(x) + y(x) - 2x = -2$$

$$y(0) = 1$$

$$y := \text{odesolve}(x, 2)$$

$$x := 0, 0.01 \dots 2$$

## 18 Алгоритм решения ОДУ второго порядка в Mathcad. Примеры.

Решение уравнения  $y'' = -y' + 2 \cdot y$        $y(0) = 1$      $y'(0) = 3$

$$y := \begin{pmatrix} 1 \\ 3 \end{pmatrix} \quad \leftarrow \text{Задание начальных условий} \quad \begin{pmatrix} y(0) \\ y'(0) \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \end{pmatrix} = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$$

$$D(t, y) := \begin{pmatrix} y_1 \\ -y_1 + 2 \cdot y_0 \end{pmatrix} \quad \begin{matrix} \leftarrow \text{Первая производная} \\ \leftarrow \text{Вторая производная} \end{matrix}$$

$$Z := \text{rkfixed}(y, 0, .5, 400, D) \quad \leftarrow \text{Вычисление решения в четырёхстах промежуточных точках на отрезке [0, 0.5].}$$

## 19 Примеры применения численных методов при создании и исследовании компьютерных моделей технических систем.

## 20 Аппроксимация и интерполяция данных, основные определения

Аппроксимация – замена исходной функции  $f(x)$  функцией  $\varphi(x)$  так, чтобы отклонение  $f(x)$  от  $\varphi(x)$  в заданной области было минимальным. ( $\varphi(x)$  – аппроксимирующая функция)

Если исходная функция задана таблично, то аппроксимация называется непрерывной или дискретной.

Интерполяция – замена исходной функции так, чтобы  $\varphi(x)$  проходила через точку исходной функции (точечная аппроксимация)

Экстраполяцией называется аппроксимация вне заданной области определения исходной функции, т.е.  $x < x_0$  и  $x > x_n$ .

Локальная интерполяция проводится только для отдельных участков отрезка.

Глобальная интерполяция для всего интервала.

## 21 Метод наименьших квадратов

Критерием близости в методе наименьших квадратов является требование минимальности суммы квадратов отклонений от аппроксимирующей функции до экспериментальных точек:

$$\Phi = \sum_{i=1}^n (y_i - f(x_i))^2 \rightarrow \min$$

Если точки расположены близко к прямой то их можно аппроксимировать зависимостью  $ax+b$ .

$$\underline{\underline{a}} := \frac{n \cdot \sum_{i=1}^n (x_i \cdot y_i) - \sum_{i=1}^n x_i \cdot \sum_{i=1}^n y_i}{n \cdot \sum_{i=1}^n (x_i)^2 - \left( \sum_{i=1}^n x_i \right)^2} \quad \underline{\underline{b}} := \frac{1}{n} \cdot \left( \sum_{i=1}^n y_i - a \cdot \sum_{i=1}^n x_i \right)$$

## 22 Линейная и сплайновая интерполяция данных в Mathcad. Примеры

Cspline(Vx,Vy) – возвращает вектор вторых производных (Vk) при приближении в узловых точках к кубическому полиному.

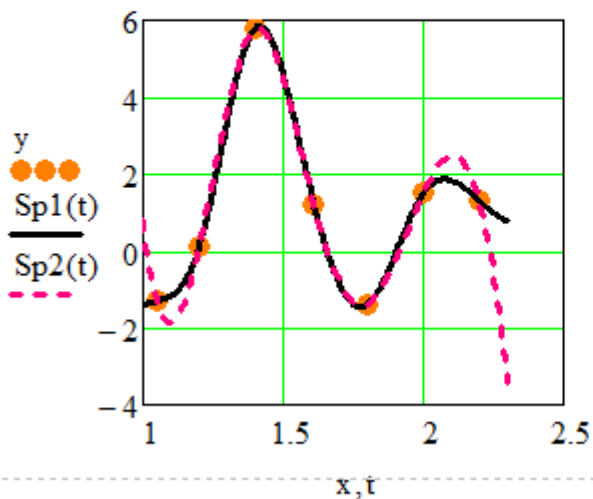
Pspline(Vx,Vy) – к параболической кривой

lspline(Vx,Vy) – в узловых точках к прямой

interp(Vk,Vx,Vy,x) – использование результата предыдущей функции

$$S1 := \text{lspline}(x,y) \quad S2 := \text{cspline}(x,y)$$

$$\text{Sp1}(Q) := \text{interp}(S1, x, y, Q) \quad \text{Sp2}(Q) := \text{interp}(S2, x, y, Q)$$



### 23 Аппроксимация данных в Mathcad по методу наименьших квадратов. Примеры

$$x := (4 \quad 4.3 \quad 4.6 \quad 4.9 \quad 5.2 \quad 5.5 \quad 5.8)^T$$

$$y := (-16 \quad -7 \quad 4 \quad 19 \quad 37 \quad 60 \quad 87)^T$$

$$F(x) := \begin{pmatrix} x^3 \\ x^2 \end{pmatrix}$$

$$Q := \text{linfit}(x, y, F) \quad Q = \begin{pmatrix} 1.987 \\ -8.946 \end{pmatrix}$$

$$G(x) := F(x) \cdot Q$$

$$x1 := 3.5, 3.9 \dots 6$$

