

Лекция №6

РЕШЕНИЕ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ

Обыкновенными дифференциальными уравнениями называют такие уравнения, которые содержат одну или несколько производных от исходной формулы $y = y(x)$. Их можно записать в виде:

$$F(x, y, y', \dots, y^{(n)})$$

где x – независимая переменная

Наивысший порядок n входящий в уравнение называется **порядком дифференциального уравнения**

Уравнения 1-го и 2-го порядка

$$F(x, y, y') = 0,$$

$$F(x, y, y', y'') = 0$$

Линейным дифференциальным уравнением называется уравнение, линейное относительно исходной формулы. Линейное уравнение 1-го порядка

$$y' - x^2 \cdot y = \sin x$$

Нелинейное дифференциальное уравнение - дифференциальное уравнение (обыкновенное или с частными производными), в к-рое по крайней мере одна из производных неизвестной функции (включая и производную нулевого порядка - саму неизвестную функцию) входит нелинейно.

Решением дифференциального уравнения называется всякая функция $y = \varphi(x)$, которая после ее подстановки в уравнение превращает его в тождество.

Решить дифференциальное уравнение - найти его общий интеграл. Под общим интегралом понимается соотношение между независимой переменной, зависимой переменной и произвольными постоянными C_1, C_2, \dots, C_n , число которых равно порядку дифференциального уравнения. Общее решение (или общий интеграл) уравнения имеет вид: $y = \phi(x, C_1, C_2, \dots, C_n)$ и геометрически выражается семейством кривых. Для выделения одной кривой из этого семейства, т.е. для получения частного решения, необходимо определить произвольную постоянную, задавшись начальными условиями. В уравнениях n -ного порядка эти условия налагаются на переменную x и её производные, а именно задаются значения $y, y', \dots, y^{(n-1)}$ при $x = x_0$:

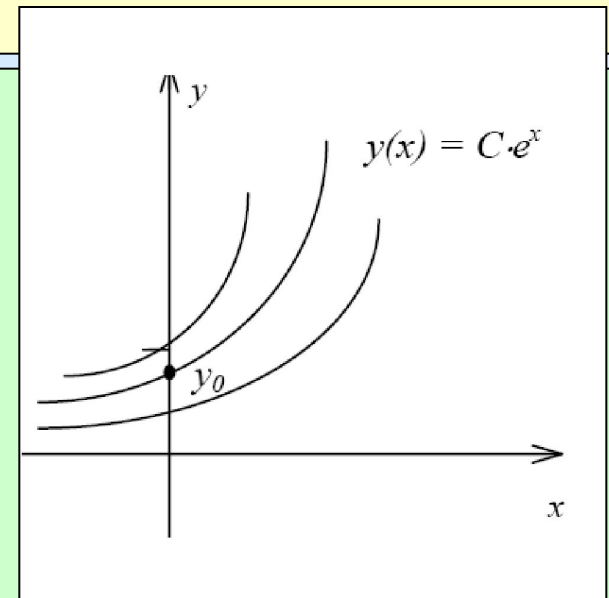
$$y_0 = y(x_0), y_0' = y'(x_0), \dots, y_0^{(n-1)} = y^{(n-1)}(x_0)$$

Выбор начального значения позволяет выделить из семейства кривых определенную кривую, т.е. найти **частное решение**.

Дифференциальное уравнение считается **разрешенным** относительно старшей производной, если оно приведено к виду

$$y^{(n)} = f(x, y, y', \dots, y^{(n-1)}).$$

Задача определения этого решения при заданных условиях называется **задачей Коши** для обыкновенного дифференциального уравнения.



Задача Коши

пусть дано дифференциальное уравнение:

$$\frac{dy}{dx} = f(x, y)$$

и начальное условие: $y(x_0) = y_0$

Требуется найти функцию $y(x)$ удовлетворяющую как указанному уравнению, так и начальному условию.

Число разработанных для задачи Коши методов очень велико.
Остановимся здесь на двух группах методов :

Одношаговые методы, в которых для нахождения следующей точки на кривой $y = f(x)$ требуется информация лишь об одном предыдущем шаге: метод Эйлера, методы Рунге - Кутты.

Методы прогноза и коррекции (многошаговые), в которых для отыскивания следующей точки кривой $y = f(x)$ требуется информация более чем об одной из предыдущих точек: метод Адамса

Метод Эйлера

Это простейший метод решения задачи Коши, позволяющий интегрировать дифференциальные уравнения первого порядка. Его точность невелика. На основе этого метода легче понять алгоритм других, более эффективных методов.

Найдём приближенное решение уравнения на отрезке $[x_0, b]$ удовлетворяющее начальному условию $y = y_0$ при $x = x_0$. Разделим отрезок $[x_0, b]$ точками $x_0, x_1, x_2, \dots, x_n = b$ на n равных частей.

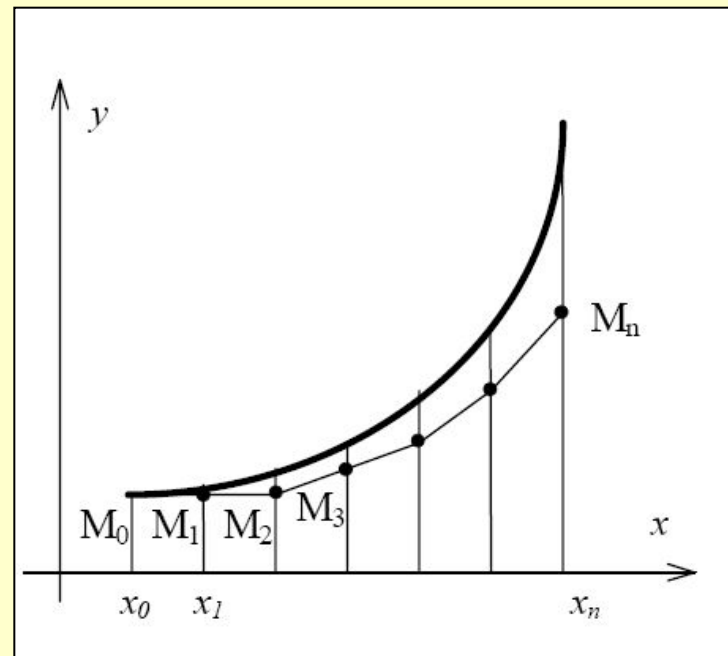
$$h = \frac{b - x_0}{n}$$

В результате на первом отрезке $[x_0, x_1]$ искомое решение приближенно представляется формулой:

$$y_1 = y_0 + f(x_0, y_0) \cdot h.$$

Иными словами на отрезке $[x_0, x_1]$ искомая интегральная кривая (точное решение) заменяется отрезком прямой M_0, M_1 касательной к кривой в точке M_0 . Тангенс угла наклона этой прямой равен $f(x_0, y_0)$. Аналогично находятся остальные приближенные значения:

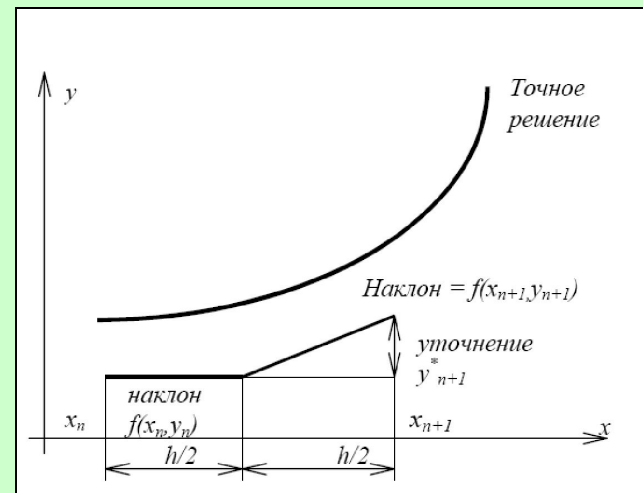
$$y_i = y_{i-1} + f(x_{i-1}, y_{i-1})h$$



Модифицированный метод Эйлера

Точность метода Эйлера можно существенно повысить, улучшив аппроксимацию производной. Это можно сделать используя среднее значение производной в начале и в конце интервала.

$$y_{n+1} = y_n + \frac{h}{2} (f(x_n, y_n) + f(x_{n+1}, y_{n+1}^*))$$



Это соотношение описывает модифицированный метод Эйлера. За повышение точности приходится расплачиваться дополнительными затратами математического времени, необходимыми для вычисления

Метод Рунге - Кутты

Наиболее распространенный метод. Метод Рунге - Кутта требует на каждом шаге четырёхкратного вычисления правой части уравнения $f(x, y)$.

Требует большого объема вычислений, однако это окупается повышенной точностью.

$$y_{i+1} = y_i + \frac{k_0 + 2 \cdot k_1 + 2 \cdot k_2 + k_3}{6}, \text{ где}$$

$$k_0 = h \cdot f(x_i, y_i)$$

$$k_1 = h \cdot f\left(x_i + \frac{h}{2}, y_i + \frac{k_0}{2}\right),$$

$$k_2 = h \cdot f\left(x_i + \frac{h}{2}, y_i + \frac{k_1}{2}\right),$$

$$k_3 = h \cdot f(x_i + h, y_i + k_2).$$

Методы прогноза и коррекции

На каждом шаге вводятся два этапа, использующие многошаговые методы:

1. С помощью **явного метода (предиктора)** по известным значениям формул в предыдущих узлах находится начальное приближение $y_{i+1} = y^{(0)}_{i+1}$ в новом узле:

$$y_{i+1} = y_i + \frac{h}{24} (55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3}).$$

2. Используется **неявный метод (корректор)** в результате итераций находится приближения

$$y_{i+1}^{(1)}, y_{i+1}^{(2)} \dots$$

$$y_{i+1} = y_i + \frac{h}{24} (9f_{i+1} - 19f_i - 5f_{i-1} + f_{i-2}).$$

Этот вариант метода прогноза и коррекции построен на основе метода Адамса четвертого порядка. Явная схема используется на каждом шаге один раз, а с помощью неявной схемы строится итерационный процесс вычисляется y_{i+1} , поскольку это значение входит в правую часть выражения $f_{i+1} = f(x_{i+1}, y_{i+1})$. Здесь необходимы значения формулы в четырех предыдущих узлах: $y_i, y_{i-1}, y_{i-2}, y_{i-3}$ необходимые при этом значения y_1, y_2, y_3 находятся по методу Рунге-Кутты, y_0 задается начальным условием. Этим характеризуется особенность многошаговых методов.

Метод Адамса

Широко распространенное семейство многошаговых методов. Простейший из них, получающийся при $k = 1$, совпадает с *методом Эйлера первого порядка точности*. В практических расчетах чаще всего используется вариант метода Адамса, имеющий четвертый порядок точности и использовавший на каждом шаге результаты предыдущих четырех. Именно его и называют обычно методом Адамса. Пусть найдены значения в четырех последовательных узлах ($k=4$). При этом имеются также вычисленные ранее значения правой части. В случае постоянного шага h конечные разности в узле x_i имеют вид:

$$\Delta f_i = f_i - f_{i-1}$$

$$\Delta^2 f_i = f_i - 2f_{i-1} + f_{i-2}$$

$$\Delta^3 f_i = f_i - 3f_{i-1} + 3f_{i-2} - f_{i-3}$$

тогда разностная сумма четвертого порядка метода Адамса запишется в виде:

$$y_{i+1} = y_i + h \cdot y_i + \frac{h^2}{2} \cdot \Delta f_i + \frac{5 \cdot h^3}{12} \cdot \Delta^2 f_i + \frac{3 \cdot h^4}{8} \cdot \Delta^3 f_i.$$

Сравнивая метод Адамса с методом Рунге - Кутты той же точности, отмечаем его экономичность. Но метод Адамса неудобен тем, что невозможно начать счет лишь по известному значению y_0 . Метод Адамса не позволяет (без усложнения формул) изменить шаг h в процессе счета; этого недостатка лишены одношаговые методы.

Общая характеристика методов

Одношаговые методы

1. Чтобы получить информацию в новой точке надо иметь данные лишь в одной предыдущей точке. Это свойство можно назвать «самостартованием».
2. Все одношаговые методы не требуют действительного вычисления производных - вычисляется лишь сама функция, однако могут потребоваться ее значения в нескольких промежуточных точках. Это влечет за собой, конечно дополнительные затраты времени и усилий.
3. Свойство «самостартования» позволяет легко менять величину шага h .

Многошаговые методы

1. Для реализации метода прогноза и коррекции необходимо иметь информацию о нескольких предыдущих точках: другими словами, они не относятся к числу «самостартующих» методов. Для получения исходной информации приходится прибегать к какому-либо одношаговому методу.
2. Одношаговые методы и методы прогноза и коррекции имеют приблизительно одинаковую точность расчета
3. Методы прогноза и коррекции требуют меньше машинного времени, но большего объема памяти компьютера, чем одношаговые.

Краевые задачи

Рассмотрим, например, линейное дифференциальное уравнение второго порядка:

$$y'' + p(x)y' + q(x)y = f(x)$$

Краевая задача состоит в отыскании решения $y = y(x)$ уравнения на отрезке $[a, b]$ удовлетворяющего на концах отрезка условиям $y(a) = A$, $y(b) = B$.

Основными методами решения являются методы в которых используется конечно-разностная форма дифференциального уравнения. Достоинство - позволяют свести решение краевой задачи к решению системы алгебраических уравнений. $y'' = f(x, y, y')$, при $y(a) = A$, $y(b) = B$ интервал $[a, b]$ можно разделить на n равных частей: $x_i = x_0 + i \cdot h$, $i = 1, 2, \dots, n$, где

$$x_0 = a, x_n = b, h = \frac{b-a}{n}$$

В точках x_i называемых узлами, стремятся получить решения y_i .

Зная координаты узлов и пользуясь разностными выражениями производных

$$y'(x_i) = \frac{1}{2h}(y_{i+1} - y_{i-1})$$
$$y''(x_i) = \frac{1}{h^2}(y_{i+1} - 2y_i + y_{i-1})$$

Подставляя эти выражения получаем систему уравнений.

$$F(x_i, y_{i-1}, y_i, y_{i+1}) = 0 \quad i = 1, 2, \dots, n-1,$$

являющуюся системой $n - 1$ алгебраических уравнений относительно сеточной функции y_1, y_2, \dots, y_{n-1} . Входящие в данную систему y_0 (при $i = 1$) и y_n (при $i = n-1$) берутся из граничных условий.

Реализация методов с использованием формул на примере

Δάφειοü äèðòáðáíöèèäëüííá óðàáíáíèà $y' = 2x^2 + 2y$

$$f(X, Y) := 2 \cdot X^2 + 2 \cdot Y$$

íà-äëüíúá òñèíäèÿ

$n := 10$ $i := 0..n$

$x_0 := 0$ $y_0 := 1$ $x_n := 1$

$$h := \frac{x_n - x_0}{n} \quad x_i := x_0 + i \cdot h$$

íàðíä Ðóíää-Êóððà

$Yrk_0 := y_0$

$$k1(f, X, Y, h) := h \cdot f(X, Y)$$

$$k2(f, X, Y, h) := h \cdot f\left(X + \frac{h}{2}, Y + \frac{k1(f, X, Y, h)}{2}\right)$$

$$k3(f, X, Y, h) := h \cdot f\left(X + \frac{h}{2}, Y + \frac{k2(f, X, Y, h)}{2}\right)$$

$$k4(f, X, Y, h) := h \cdot f(X + h, Y + k3(f, X, Y, h))$$

$$RK(f, X, Y, h) := \frac{1}{6} \cdot (k1(f, X, Y, h) + 2 \cdot k2(f, X, Y, h) + 2 \cdot k3(f, X, Y, h) + k4(f, X, Y, h))$$

$$Yrk_{i+1} := Yrk_i + RK(f, x_i, Yrk_i, h)$$

íàðíä Ýéèáðà

$$y_{i+1} := y_i + h \cdot f(x_i, y_i)$$

íàðíä Äääíñà

$j := 0..3$

$Y_{Adams_j} := Yrk_j$ $t := 3..n$

$$Y_{Adams_{t+1}} := Y_{Adams_t} + h \cdot f(x_t, Y_{Adams_t}) + \frac{1}{2} \cdot h \cdot (f(x_t, Y_{Adams_t}) - f(x_{t-1}, Y_{Adams_{t-1}})) + \bullet$$

$$\frac{5}{12} \cdot h \cdot (f(x_t, Y_{Adams_t}) - 2 \cdot f(x_{t-1}, Y_{Adams_{t-1}}) + f(x_{t-2}, Y_{Adams_{t-2}})) + \bullet$$

$$\frac{3}{8} \cdot h \cdot (f(x_t, Y_{Adams_t}) - 3 \cdot f(x_{t-1}, Y_{Adams_{t-1}}) + 3 \cdot f(x_{t-2}, Y_{Adams_{t-2}}) - f(x_{t-3}, Y_{Adams_{t-3}}))$$

Сравнение методов решения ОДУ

Нужно сравнить результаты решения ОДУ с помощью методов Рунге-Кутты и Адамса:

$y_i =$ $Y_{rk}_i =$ $Y_{Adams}_i =$ $Y_{toch}(x_i) =$

1
1.2
1.442
1.738
2.104
2.557
3.118
3.814
4.675
5.738
7.047

1
1.222
1.498
1.843
2.278
2.827
3.52
4.393
5.489
6.864
8.583

1
1.222
1.498
1.843
2.278
2.827
3.519
4.391
5.486
6.86
8.577

1
1.222
1.498
1.843
2.278
2.827
3.52
4.393
5.49
6.864
8.584

$$\frac{|Y_{toch}(x_{10}) - y_{10}|}{Y_{toch}(x_{10})} = 17.899\%$$

$$\frac{|Y_{toch}(x_{10}) - Y_{rk_{10}}|}{Y_{toch}(x_{10})} = 2.35556 \times 10^{-3} \%$$

$$\frac{|Y_{toch}(x_{10}) - Y_{Adams_{10}}|}{Y_{toch}(x_{10})} = 0.074\%$$

Результаты получены с помощью Mathcad

Дифференциальные уравнения 1-го порядка

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

y - вектор начальных условий размерности n , где n -порядок ДУ или число уравнений в системе (если решается система уравнений).

$x1$, $x2$ - граничные точки интервала, на котором ищется решение ДУ.

$npoints$ - число точек (не считая начальной точки), в которых ищется приближенное решение.

$D(x, y)$ - функция возвращающее значение в виде вектора из n элементов, содержащих первые производные неизвестных функций.

Функция *rkfixed* использует метод Рунге-Кутты 4-го порядка.

В результате решения получается матрица, имеющая два следующих столбца:

- 1-ый столбец содержит точки, в которых ищется решение ДУ.
- 2-ой столбец содержит значения найденного решения в соответствующих точках.

Пример Решение уравнения $y' + 3y = 0$.

$y_0 := 4$ - задание начального значения.

$D(x, y) := -3y_0$ - Определение функции, задающей производную: $y' = -3y$

$Z := \text{rkfixed}(y, 0, 4, 100, D)$ - вычисление решения в 100 промежутках

$I := 0 \dots \text{rows}(Z) - 1$

Функция *rkfixed* использует универсальный метод для решения дифференциальных уравнений. Хотя этот метод не всегда самый быстрый, но он почти всегда приводит к искомому результату. Однако, есть случаи, в которых можно использовать другую функцию MathCAD для решения ДУ. Эти случаи относятся к 3-м широким категориям задач:

решается не задача с начальными условиями (Коши), а краевая задача

система уравнений может иметь некоторые специфические свойства, используя которые, можно решить её более точно для решения систем с гладкими решениями (*Bulstoer*) или медленно изменяющимися решениями (*Rkadapt*)

требуется найти решения только в одной фиксированной точке (*bulstoer, rkadapt*).

Медленно изменяющиеся решения

Задавшись фиксированным числом точек, можно аппроксимировать функцию более точно, если вычислить её значение в точках, расположенных следующим образом: часто на тех интервалах, где функция меняется быстро, и не очень часто, где функция изменяется менее быстро. Если известно, что искомое решение достаточно гладкое, то лучше использовать функцию *Rkadapt* для поиска приближенного решения. Функция *Rkadapt* проверяет, как быстро изменяется приближенное решение, и адаптирует соответственно размер шага. Хотя *Rkadapt* при решении ДУ использует во внутренних расчетах переменный шаг, возвращает приближенное решение на равномерной сетке (в равноотстоящих точках).

Нахождение приближенного решения только в конечной точке

Часто возникает задача, когда необходимо найти приближенное значение только в конечной точке интервала $y(x)$. Хотя функции, описанные выше, будут давать значения $y(x_2)$, при этом они будут делать много ненужной дополнительной работы, возвращая промежуточные значения $y(x)$.

bulstoer($y, x_1, x_2, acc, D, kmax, save$)

rkadapt($y, x_1, x_2, acc, D, kmax, save$),

acc . параметр, контролирующий точность решения. Малое значение *acc* вынуждает алгоритм брать меньше шаги вдоль траектории, что повышает точность решения ($acc=0,001$).

Решение дифференциального уравнения первого порядка

$y' + 10y - 4x = 0$ с начальными условиями $y(0) = 0.5$

$y_0 := 0.5$ - задание начального значения. Для создания индекса нажмите клавишу [.

$D(x, y) := -10 \cdot y_0 + 4 \cdot x$ - определение функции, задающей производную: $y' = -10y + 4x$

Вычисление решения на отрезке [0,1]:

С помощью функции rkfixed

$Z := \text{rkfixed}(y, 0, 1, 5, D)$

$i := 0.. \text{rows}(Z) - 1$

С помощью функции Rkadapt

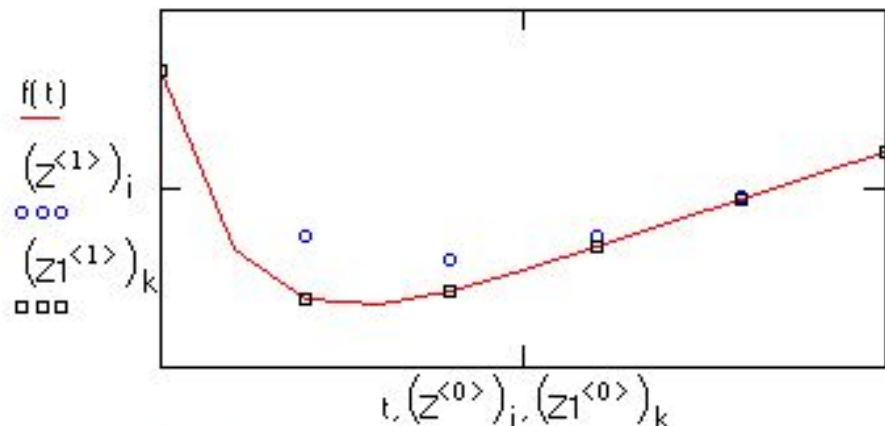
$Z1 := \text{Rkadapt}(y, 0, 1, 5, D)$

$k := 0.. \text{rows}(Z1) - 1$

Точное решение

$t := 0, 0.1.. 1$

$f(t) := \frac{-1}{25} + \frac{2}{5} \cdot t + \frac{1}{2} \cdot e^{-10 \cdot t} + \frac{1}{25} \cdot e^{-10 \cdot t}$



x	y(x)
0	0.5
0.2	0.22
0.4	0.18
0.6	0.22
0.8	0.287
1	0.362

— Точное решение
 ○○ rkfixed
 □□ Rkadapt

Решение краевых задач

В этом случае Mathcad предлагает использовать функцию $sbval$, чтобы найти недостающие начальные условия в точке a .

Sbval($v, a, b, D, load, score$)

Возвращает вектор, содержащий недостающие начальные условия в точке a . Вектор v задает начальные приближения, a, b - граничные точки интервала решений, $D(x, y)$ - функция-вектор с первыми производными неизвестных функций. $load(a, v)$ - функция-вектор, возвращающая значение начальных условий в точке a . $score(b, y)$ - функция-вектор, каждый элемент которого содержит разность между начальным условием заданным в точке b , и значением искомого решения в этой точке. После того, как эти недостающие начальные условия будут получены, можно решать обычную задачу с начальными условиями - задачу Коши, используя любую из функций, описанных выше

Решение дифференциального уравнения:

$$y'' = 2x + 3y \text{ с граничными условиями } y(0) = 0, y(1) = 1$$

1. Преобразование краевой задачи к задаче Коши

$v_0 := 1$ - стартовое значение для поиска величины $y'(0)$.

$$\text{load}(a, v) := \begin{pmatrix} 0 \\ v_0 \end{pmatrix} \begin{array}{l} \text{- известное } y(0) \\ \text{- неизвестное начальное} \\ \text{условие. Должно быть} \\ \text{найдено с помощью sbval.} \end{array}$$

$$D(x, y) := \begin{pmatrix} y_1 \\ 2 \cdot x + 3 \cdot y_0 \end{pmatrix} \begin{array}{l} \text{- первая производная} \\ \text{- вторая производная} \end{array}$$

$\text{score}(b, y) := y_0 - 1$ - разница между вычисленными и заданными значениями y .

$S := \text{sbval}(v, 0, 1, D, \text{load}, \text{score}) \quad S = 0.388$ $y'(0)$ - недостающее начальное условие, требуемое функцией rkfixed.

2. Решение задачи Коши для дифференциального уравнения второго порядка

$$y'' = 2x + 3y \text{ с начальными условиями } y(0) = 0, y'(0) = 0.388$$

$$y := \begin{pmatrix} 0 \\ 0.388 \end{pmatrix} \text{ - задание начальных условий}$$

$$D(x, y) := \begin{pmatrix} y_1 \\ 2 \cdot x + y_0 \cdot 3 \end{pmatrix} \begin{array}{l} \text{- первая производная} \\ \text{- вторая производная} \end{array}$$

$Z := \text{rkfixed}(y, 0, 1, 10, D)$ - Вычисление решения в 10 промежуточных точках на отрезке $[a, b]$

$$Z = \begin{bmatrix} x & y(x) & y'(x) \\ 0 & 0 & 0.388 \\ 0.1 & 0.039 & 0.404 \\ 0.2 & 0.082 & 0.452 \\ 0.3 & 0.131 & 0.534 \end{bmatrix}$$

+

Символьное решение линейных дифференциальных уравнений

Для получения аналитического решения линейных ОДУ в Mathcad необходимо выполнить следующие действия:

- Напечатать исходное уравнение $f(x, y)$ в виде выражения $f(x, y)$, используя оператор дифференцирования.
- Отметив независимую переменную, выполнить прямое преобразование Лапласа **Символы > Преобразование > Лапласа**. Результат для ОДУ выше 1-го порядка будет помещен в буфер обмена. Вызовите его нажав клавишу **F4**.
- По результатам преобразования Лапласа составить алгебраическое уравнение, введя обозначения $L = \text{laplace}(y(t), t, s)$, $C1 = y(0)$ и $C2 = \text{diff}(y(0), 0)$.
- Решить составленное уравнение относительно переменной L , используя команду **Символы > Переменные > Вычислить**.
- Отметить переменную s и произведя обратное преобразование Лапласа **Символы > Преобразование > Лапласа Обратное** получить решение заданного ОДУ.

Символьное решение ОДУ 1-го порядка

$$\frac{d}{dt}y(t) - k \cdot y(t) = m \cdot t$$

$$\frac{d}{dt}y(t) - k \cdot y(t) = m \cdot t$$

Отметив переменную t , выполним команду
Символы \Rightarrow **Преобразование** \Rightarrow **Лапласа**

$$s \cdot \text{laplace}(y(t), t, s) - y(0) - k \cdot \text{laplace}(y(t), t, s) = \frac{m}{s^2}$$

Введя обозначения
 $L = \text{laplace}(y(t), t, s)$, $C1 = y(0)$ имеем: +

$$L \cdot s - C1 - k \cdot L = \frac{m}{s^2}$$

Решая это уравнение относительно L с помощью
команды **Символы** \Rightarrow **Переменные** \Rightarrow **Вычислить** получим:

$$\frac{-(C1 \cdot s^2 + m)}{[s^2 \cdot (-s + k)]}$$

Отметив переменную s , выполним команду
Символы \Rightarrow **Преобразование** \Rightarrow **Лапласа Обратное**
и найдем временную зависимость - решение заданного уравнения:

$$\frac{-1}{k^2} \cdot m - \frac{1}{k} \cdot m \cdot t + \exp(k \cdot t) \cdot C1 + \frac{1}{k^2} \cdot \exp(k \cdot t) \cdot m$$

КОНЕЦ ЛЕКЦИИ!