

Розв'язок систем рівнянь за допомогою блоку Given-Find

Розв'язок систем рівнянь Mathcad здійснює чисельними методами. При цьому повинно бути задано деяке початкове наближення для тих змінних, значення яких необхідно знайти. Ґрунтуючись на цих початкових даних, Mathcad послідовно уточнюватиме розв'язок до тих пір, поки не підбере найточніші значення. Проблеми виникають, коли нелінійна система має декілька розв'язків. За один раз Mathcad знаходить тільки один розв'язок, який звичайно більш близький до заданого початкового наближення. Тому в таких випадках необхідно вирішувати систему кілька разів з різними початковими наближеннями.

Обчислювальний блок складається з декількох компонент, наступних на листі в *строго певному порядку*:

1. Початкове наближення (присвоєння початкових значень змінним).
2. Команда *Given*, яку необхідно набрати з клавіатури.
3. Рівняння, які необхідно вирішити. Рівняння вводяться в звичайній математичній формі, але замість простого знака рівності «=» використовується оператор логічної рівності (вводиться шляхом натиснення *Ctrl-=*).
4. Звернення до функції *Find*. Аргументами функції є імена змінних, щодо яких розв'язується система. Функція повертає вектор значень, де перший елемент відповідає першій змінній в списку аргументів, другий елемент - другої змінної і так далі.

Приклад.

Розв'яжемо систему нелінійних рівнянь:

$$\begin{cases} x^2 - x = 2; \\ x - 0,5y = 0. \end{cases}$$

Дана система має два розв'язки. Знайдемо одне з них (мал. 5) з початковим наближенням $x = 0; y = 0$.

Мал. 5. Розв'язок системи в Mathcad

$$\begin{array}{l} x := 0 \quad y := 0 \\ \\ \text{Given} \\ \\ x^2 - x = 2 \\ \\ x - 0.5y = 0 \\ \\ \text{Find}(x, y) = \begin{pmatrix} -1 \\ -2 \end{pmatrix} \end{array}$$

- Останній запис - вектор $(-1; -2)$ є значення, яке повернула функція Find, тобто одне з розв'язків системи. Знайти другий розв'язок можна, якщо узяти інше початкове наближення $x = 2; y = 2$. Тоді функція Find поверне вектор $(2; 4)$.
- Починаючи з *Mathcad 2000* існує можливість одночасно знайти декілька розв'язків. Для цього система рівнянь і початкові наближення повинні бути переписані у векторній формі (мал. 6). Кожна змінна буде вектором, що містить стільки компонент, скільки розв'язків знаходиться. В системі зміни торкнуться переважно членів з перемножуванням змінних. Припустимо, що в рівнянні присутній вираз $x \cdot x$. Якщо $x = (x_1; x_2)$ - вектор, то $x \cdot x = (x_1^2; x_2^2)$. Нам же необхідний результат поелементного перемножування. Для цього існує спеціальна операція, записувана як

Мал. 6. Приклад одночасного пошуку декількох розв'язків

$$\begin{array}{l} x := \begin{pmatrix} 0 \\ 2 \end{pmatrix} \quad y := \begin{pmatrix} 0 \\ 2 \end{pmatrix} \\ \text{Given} \\ \longrightarrow \\ (x \cdot x) - x = 2 \\ x - 0.5y = 0 \\ \begin{pmatrix} X \\ Y \end{pmatrix} := \text{Find}(x, y) \\ X = \begin{pmatrix} -1 \\ 2 \end{pmatrix} \quad Y = \begin{pmatrix} -2 \\ 4 \end{pmatrix} \end{array}$$

- Зміни торкнулися і частини отримання результату. В даному випадку функція Find поверне вектор з двох елементів, які ми позначили як X і Y. Відповідно перше розв'язок - (-1; -2); друге розв'язок - (2; 4).

Аналітичний розв'язок лінійних і нелінійних систем рівнянь

- Даний алгоритм використовується для отримання розв'язків в загальному вигляді. Звичайно при цьому система рівнянь записується тільки з використанням буквених позначень змінних, без конкретних чисел. Для отримання аналітичного розв'язок (мал. 7, 8) використовується оператор аналітичного обчислення « \approx » замість оператора числового обчислення « $=$ ».

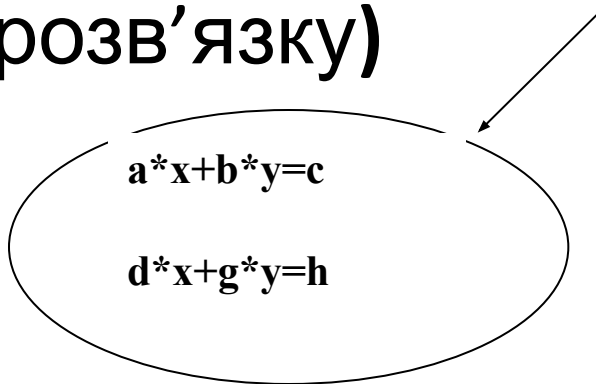
Мал. 7. Приклад аналітичного розв'язку нелінійної системи

Given

$$y = \cos(\omega \cdot x + t)$$
$$y = \sin(\omega \cdot x - t)$$

Find(x, y) \rightarrow $\left(\begin{array}{c} \frac{1}{4} \cdot \frac{\pi}{\omega} \\ \cos\left(\frac{1}{4} \cdot \pi + t\right) \end{array} \right)$

- Вигляд системи рівнянь (не елемент розв'язку)



$a*x+b*y=c$
 $d*x+g*y=h$

Робота з послідовностями. Обчислення елементів матриць

- Mathcad дозволяє задавати значення елементів матриць не тільки безпосередньо, але і шляхом визначення загальних законів обчислення елементів. При цьому виникає необхідність визначити границь дії законів, тобто діапазону зміни індексів матриць. Для цього широко використовуються послідовності.
- *Зауваження:* в Mathcad нумерація елементів векторів і матриць починається з нуля.
- *Приклад.* Чисельний ряд Фібоначчі визначається за наступними правилами:
 - $F1 = 1; F2 = 1;$
 - $F_i = F_{i-1} + F_{i-2}; i > 2.$
- Даний ряд нескінченний. Але ми не можемо визначити в Mathcad нескінченні послідовності, можливе лише знаходження частини ряду. Тому замість вказівки $i > 2$ необхідно задати конкретний діапазон зміни індексу i , наприклад нас можуть цікавити члени ряду з третього по восьмий. Розв'язок показано на мал. 9.

Мал. 9. Обчислення частини чисельного ряду Фібоначчі в Mathcad

```
i := 3..8  
  
F1 := 1   F2 := 1  
  
Fi := Fi-1 + Fi-2  
  
Fi =  


|    |
|----|
| 2  |
| 3  |
| 5  |
| 8  |
| 13 |
| 21 |


```

- Тут спочатку визначений діапазон зміни індексу i , потім задані значення двох перших членів ряду і записана загальна формула для обчислення члена з індексом i . Остання частина (стовпець значень) є частина ряду $F3 \dots F8$, обчислена системою. Такий порядок запису виразів пояснюється тим, що Mathcad обчислює формули послідовно, в порядку «зліва направо і зверху вниз». При обчисленні члена $F3$ по заданій рекурентній формулі системі вже винні бути відомі реальні значення $F1$ і $F2$. Тому вони повинні бути записані до цієї формули.
- *Приклад.* Розглянемо чисельну інтеграцію явним методом Ейлера.
- Спочатку стисло розглянемо суть методу. Нехай дано диференціальне рівняння в загальному вигляді:

$$\frac{dx}{dt} = F(x, t), \quad t \in [t_0, T], \quad x(t_0) = x_0$$

- Необхідно проінтегрувати дане рівняння. Якщо функція $F(x,t)$ достатньо складна, то таке рівняння може не мати очевидного аналітичного розв'язок. В таких випадках необхідно використовувати наближені чисельні методи, до яких і відноситься метод Ейлера.
- Перш за все, необхідно замінити похідну яким-небудь наближеним співвідношенням. Skorистаємося визначенням похідної:

$$\frac{dx}{dt} = \lim_{\Delta t \rightarrow 0} \frac{\Delta x}{\Delta t}$$
- На жаль, комп'ютери не можуть безпосередньо працювати з нескінченно малими величинами. Тому перейдемо до кінцевих величин:
- Така проста формула буде тим точнішою, чим менше Δt . У будь-якому випадку в розв'язок вноситься погрішність, яка робитиме більш менш істотний вплив на відповідність результату точному рішення. Розгляд цього питання виходить за рамки даної допомоги. В першому наближенні можна рекомендувати вибирати в діапазоні $10^{-4} \leq \Delta t \leq 10^{-3}$.
- Далі вводиться дискретний час:
- де Δt - крок інтегрування за часом; i - номер кінцевого кроку інтегрування. Функцію $F(x,t)$ можна обчислювати як в точці (x_i, t_i) , так і в точці (x_{i+1}, t_{i+1}) - це не впливає на точність методу, хоча може впливати на характер зростання погрішності.

$$t_{i+1} = t_i + \Delta t, \quad t_i = 0, \Delta t = 0, \dots, N, \quad N = T/\Delta t$$

- Перший варіант відповідає явному методу Ейлера, другий - неявному.

- Одержуємо вираз $\frac{\Delta x}{\Delta t} = \frac{x_{i+1} - x_i}{\Delta t} = F(x_i, t_i)$

- ,
- з якого виходить розрахункова формула явного методу Ейлера



- Розрахуємо перехідний процес в електричному ланцюзі:

- Враховуючи, що тут сума напруг на резисторі і конденсаторі рівна нулю, а струм, що протікає через резистор, рівний струму через конденсатор, тобто, можна вивести диференціальне рівняння

- $\frac{(U_c)_{i+1} - (U_c)_i}{h} = -\frac{1}{RC} (U_c)_i$ $(U_c)_0 = 1$ $(U_c)_{i+1} = (U_c)_i - \frac{h}{RC} (U_c)_i$
- Нехай $R = 1, 3 \in \mathbb{R}$ з початковою умовою $U_c(0) = 1$.
Введемо дискретний час: ;

Мал. 10. Дві форми запису розв'язку в Mathcad

$$\begin{aligned}
 &N := 5 \quad i := 0..N \\
 &h := 0.1 \\
 &R := 1 \quad C := 1 \\
 &t_0 := 0 \quad U_{c_0} := 1 \\
 &t_{i+1} := t_i + h \quad U_{c_{i+1}} := U_{c_i} - \frac{h}{R \cdot C} \cdot U_{c_i} \\
 &t = \begin{pmatrix} 0 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \\ 0.5 \\ 0.6 \end{pmatrix} \quad U_c = \begin{pmatrix} 1 \\ 0.9 \\ 0.81 \\ 0.729 \\ 0.656 \\ 0.59 \\ 0.531 \end{pmatrix}
 \end{aligned}$$

$$\begin{aligned}
 &N := 5 \quad i := 0..N \\
 &h := 0.1 \\
 &R := 1 \quad C := 1 \\
 &t_0 := 0 \quad U_{c_0} := 1 \\
 &\begin{pmatrix} t_{i+1} \\ U_{c_{i+1}} \end{pmatrix} := \begin{pmatrix} t_i + h \\ U_{c_i} - \frac{h}{R \cdot C} \cdot U_{c_i} \end{pmatrix} \\
 &t = \begin{pmatrix} 0 \\ 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \\ 0.5 \\ 0.6 \end{pmatrix} \quad U_c = \begin{pmatrix} 1 \\ 0.9 \\ 0.81 \\ 0.729 \\ 0.656 \\ 0.59 \\ 0.531 \end{pmatrix}
 \end{aligned}$$

- Форма (а) більш проста, але не дозволяє записувати розв'язок систем диференціальних рівнянь, праві частини яких є взаємозалежними. Друга форма (б) більш універсальна.
- Одержаний вектор U_c містить значення $U_c(t_i)$ для заданих дискретних моментів часу. Можна легко переконатися, що одержаний чисельний схожий на точний аналітичний.

$$U_c(t) = e^{-\frac{1}{R \cdot C} t}$$

$$R_i := e^{-\frac{1}{R \cdot C} t_i}$$

$$R = \begin{pmatrix} 1 \\ 0.905 \\ 0.819 \\ 0.741 \\ 0.67 \\ 0.607 \end{pmatrix}$$

- Додамо такі вирази:
- і порівняємо значення R (точний розв'язок)
- В Mathcad існують спеціальні стандарти інтегрування диференціальних рівнянь (функція `rkfixed`, блок `Given-Odesolve`), які розглянемо далі.

Розв'язок задач нецілочисельної оптимізації

- Задача *нецілочисельної оптимізації* полягає в тому, щоб підібрати такі значення аргументів цільової функції, за яких ця функція приймає екстремальне (мінімальне або максимальне) значення і дотримується задана система обмежень на значення аргументів. В математичній формі задача може бути сформульована таким чином:

$$\text{op}_i \in \{>, <, \leq, \geq, =, \neq\},$$

- де f - цільова функція; n - кількість аргументів цільової функції; g_i, h_i - елементи системи обмежень (рівності або нерівностей різного вигляду); d_o - кількості обмежень.

Приклад задачі оптимізації

- Припустимо, що якомусь студенту необхідно скласти два заліки в один день. Він поставив собі задачу здати їх якнайкраще, причому так, щоб оцінка за кожний залік була не менше 2,5 балів. У нього залишилося 24 години. Студент припустив, що, витративши одну годину на перший предмет, він підвищить можливу оцінку на 0,5 бали, витративши ту ж годину на другий предмет, - підвищить оцінку по ньому на 0,25 бали. Необхідно визначити, скільки годин (x) йому треба витратити на перший предмет і скільки (y) на другий предмет, щоб виконати поставлену задачу. Математично задача запишеться так:
$$2,5 \leq 0,5x \leq 5;$$
$$2,5 \leq 0,25y \leq 5;$$
$$x + y \leq 24$$
- (сума оцінок з предметів)
- і повинні виконуватися такі обмеження:

- У системі Mathcad такі задачі розв'язуються за допомогою блоків Given-Maximize і Given-Minimize. Так само, як і при розв'язку систем рівнянь, вирішальний блок складається з декількох компонент, наступних на листі (мал. 14) в *строго визначеному порядку*:
 - 1. Присвоєння початкових значень змінним, щодо яких розв'язується задача оптимізації.
 - 2. Визначення цільової функції.
 - 3. Команда *Given*.
 - 4. Обмеження, записувані в звичайній математичній формі. Можуть використовуватися всі вказані вище знаки відносин, але замість простого знака рівності «=» використовується оператор логічної рівності (вводиться шляхом натиснення *Ctrl-=*). *Зауваження*: система Mathcad при мінімізації і максимізації сприймає знаки строгої нерівності (< >) як знаки нестрогої нерівності .
 - 5. Звернення до однієї з функцій *Minimize* або *Maximize* для відповідно мінімізації або максимізації. Першим аргументом завжди є ім'я цільової функції. Далі слідують імена змінних, щодо яких розв'язується задача. Функція повертає вектор значень, де перший елемент відповідає першій змінній в списку аргументів, другий елемент - другої змінної і так далі.

Мал. 14. Розв'язок задачі про студента в системі Mathcad

$$\begin{aligned}x &:= 1 & y &:= 2 \\f(x, y) &:= 0.5 \cdot x + 0.25 \cdot y \\ \text{Given} \\ &0.5 \cdot x \geq 2.5 & 0.5 \cdot x &\leq 5 \\ &0.25 \cdot y \geq 2.5 & 0.25 \cdot y &\leq 5 \\ &x + y &\leq 24 \\ \text{Maximize}(f, x, y) &= \begin{pmatrix} 10 \\ 14 \end{pmatrix}\end{aligned}$$

Чисельний розв'язок диференціальних рівнянь

- Розглянемо стандартні засоби чисельного розв'язку диференціальних рівнянь і систем диференціальних рівнянь.
- **Чисельний розв'язок одного диференціального рівняння**
- Mathcad 2000 дозволяє без додаткових перетворень чисельно вирішити диференціальне рівняння, явно дозволене щодо старшої похідної (мал. 15).

- $$\frac{d^2}{dt^2}x(t) - \sin\left(\frac{d}{dt}x(t)\right) = -x(t)$$

$$\cos\left(\frac{d^2}{dt^2}x(t)\right) = -x(t)$$

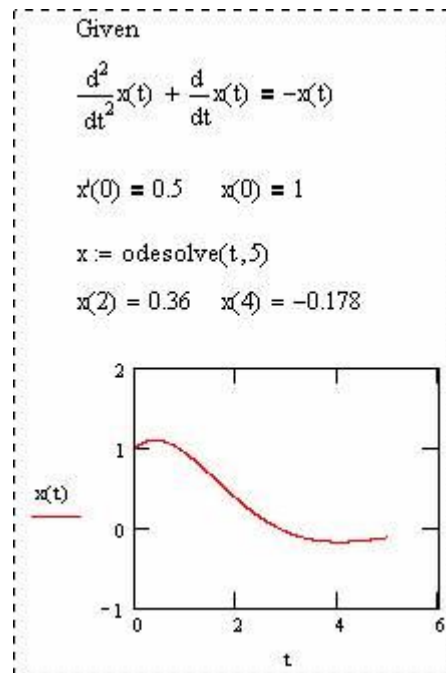
- a)

б)

- Розв'язок здійснюється за допомогою спеціального блоку *Given-Odesolve*, що складається з таких компонент:
 1. Команда *Given*.
 2. Диференціальне рівняння, записане в традиційній математичній формі з такими особливостями: а) замість простого знака рівності «=» використовується оператор логічної рівності (вводиться натисненням *Ctrl-=*); б) при позначенні функції, що інтегрується, завжди вказується аргумент (тобто замість функції $x(t)$ не можна писати просто x); в) при запису похідних використовуються або стандартні оператори i , або ставляться (за допомогою *Ctrl-F7*) символи похідної, наприклад, $x'(t)$, $x''(t)$.
 3. Вказуються початкові або кінцеві значення функції, що інтегрується, і її похідних (за винятком більшого порядку), входять в рівняння. Значення вводяться у традиційній формі з використанням оператора логічної рівності. Число значень повинне співпадати з порядком рівняння. Для рівняння другого
 - порядку вигляду повинні бути задані початкові значення функції і її першої похідної, наприклад $x(0) = 1$; $x'(0) = 0,5$. Для введення символу похідної «'» використовується комбінація клавіш *Ctrl-F7*.
 4. Звернення до функції *Odesolve*. Перший аргумент - завжди ім'я незалежної змінної. Другий аргумент - кінцеве значення незалежної змінної. Третій (необов'язковий) аргумент - кількість проміжних точок розв'язок. *Odesolve* повертає функцію, що представляє наближений (чисельний) розв'язок диференціального рівняння в заданому інтервалі часу. Така функція може бути використана для визначення значень функції, що інтегрується, в різних точках, а також для побудови графіка.

Вирішимо вищезгадане диференціальне рівняння при значеннях $t = 0..5$;


знайдемо значення x при $t = 2; 4$, та побудуємо графік розв'язку.



Лекція 6 Символьні обчислення та програмування в середовищі "Mathcad"

Символьні обчислення.

В *MathCad* є широкий спектр обчислень в символному вигляді. Для обчислення в символному вигляді натисненням кнопки

 на панелі математичних знаків відкрийте вкладку *Символіка*, як це показано на рис.13.

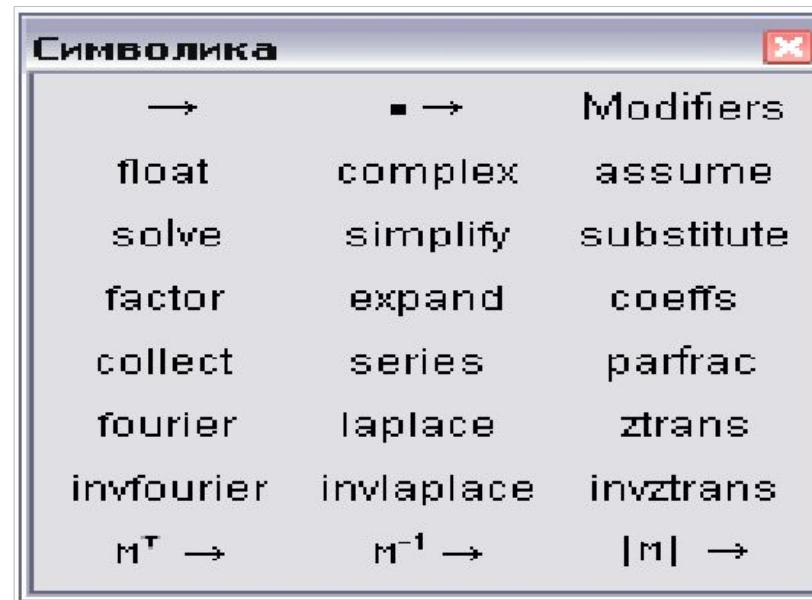


Рис.13

Найпростішим прикладом символьних обчислень є обчислення невизначених інтегралів. Для цього з використанням символу невизначеного інтеграла на панелі *Числення* наберіть необхідний невизначений інтеграл. Потім замість символу $=$ наберіть символічний знак рівності $>$ на панелі *Оцінка* і натисніть **[Enter]**. Знак $>$ також вводиться комбінацією клавіш **[Ctrl+?]**.

Приклад символьного обчислення інтеграла показаний на рис.14. На цьому ж малюнку показані приклади символьного обчислення потрійного інтеграла і певного інтеграла з параметрами *a, b*.

Відмітимо, що далеко не всі інтеграли і тим більше подвійні і потрійні *Mathcad* може обчислити в символьному вигляді. Приклад необчислюваного в символьному вигляді інтеграла (інтеграл, що не береться) наведений на рис. 14.

Аналогічним чином в символьному вигляді можна обчислити похідні будь-якого порядку, суми, твору. Приклад таких обчислень наведений на рис. 15.

$$\int \sin(x) \cdot e^x dx \rightarrow \frac{-1}{2} \cdot \exp(x) \cdot \cos(x) + \frac{1}{2} \cdot \sin(x) \cdot \exp(x)$$

$$\int \int \int x^2 \cdot \sin(y) \cdot e^z dx dy dz \rightarrow \frac{-1}{3} \cdot \cos(y) \cdot x^3 \cdot \exp(z)$$

$$\int_0^a \int_{x \left(\frac{b}{a}\right)}^b (a-x)^2 + (b-y)^2 dy dx \rightarrow \frac{1}{4} \cdot a^3 \cdot b + \frac{1}{12} \cdot a \cdot b^3$$

$$\int \frac{x^2}{e^x + \sin(x)} dx \rightarrow \int \frac{x^2}{(\exp(x) + \sin(x))} dx$$

Рис.14

Mathcad дає можливість спрощення виразу *Simplify*, розкладання виразу за ступенями *Expand* і розкладання виразу на множники *Factor*. Операція $\>$ має на увазі *Symplify* за умовчанням, проте її безпосереднє використання дає більш простий результат, як це показано на рис.15 для добутків.

$$\sum_i i^2 \rightarrow \frac{1}{3} \cdot i^3 - \frac{1}{2} \cdot i^2 + \frac{1}{6} \cdot i$$

$$\sum_k \frac{1}{(2 \cdot k - 1)^2} \rightarrow \frac{-1}{4} \cdot \text{Psi}\left(1, k - \frac{1}{2}\right)$$

$$\prod_k \frac{k^2}{k-1} \text{ simplify} \rightarrow \Gamma(k) \cdot (k-1)$$

Рис.15

Для виконання відповідної команди типу **Simplify**, **Factor** або **Expand** необхідно замість > вибрати відповідне слово на панелі *Символіка*. Приклад застосування команд **Simplify**, **Expand**, **Factor** наведений на рис.16.

$$\sin(x)^2 + \cos(x)^2 \text{ simplify} \rightarrow 1 \quad \frac{a^2 - b^2}{(a + b) \cdot (a - b)} \text{ simplify} \rightarrow 1$$

$$\frac{-5}{x} + \frac{5}{x-1} - \frac{5}{(x-1)^2} + \frac{6}{(x-1)^3} - \frac{4}{(x-1)^4} \text{ simplify} \rightarrow \frac{(x^2 - 5)}{[x \cdot (x-1)^4]}$$

$$\sin(5 \cdot x) \text{ expand,2} \rightarrow 16 \cdot \sin(x) \cdot \cos(x)^4 - 12 \cdot \sin(x) \cdot \cos(x)^2 + \sin(x)$$

$$(a + b)^5 \text{ expand,2} \rightarrow a^5 + 5 \cdot a^4 \cdot b + 10 \cdot a^3 \cdot b^2 + 10 \cdot a^2 \cdot b^3 + 5 \cdot a \cdot b^4 + b^5$$

$$3 - (4 \cdot \cos(2 \cdot a) + \cos(4 \cdot a)) \text{ expand,2} \rightarrow 2 - 4 \cdot \cos(2 \cdot a) - 8 \cdot \cos(a)^4 + 8 \cdot \cos(a)^2$$

$$x^2 - y^2 \text{ factor,2} \rightarrow (x - y) \cdot (x + y) \quad [(a^2) - 2 \cdot a \cdot b + b^2] \text{ factor,2} \rightarrow (a - b)^2$$

$$\sum_n x - n \text{ factor,2} \rightarrow n \cdot (x - 1) \quad x^3 - 1 \text{ factor,2} \rightarrow (x - 1) \cdot (x^2 + x + 1)$$

Рис.16

Команда **Collect** розкладає вираз по ступенях вказаної в цій команді змінної, якщо таке уявлення можливе. Приклад використання команди **Collect** наведений на рис. 17.

$$(a + b)^5 \text{ collect, } a \rightarrow a^5 + 5 \cdot a^4 \cdot b + 10 \cdot a^3 \cdot b^2 + 10 \cdot a^2 \cdot b^3 + 5 \cdot a \cdot b^4 + b^5$$

$$(x - a) \cdot (x - b) \cdot (x - c) \text{ collect, } x \rightarrow x^3 + (-a - b - c) \cdot x^2 + [a \cdot b - (-a - b) \cdot c] \cdot x - a \cdot b \cdot c$$

$$(a + b + c)^2 \text{ collect, } a \rightarrow a^2 + (2 \cdot b + 2 \cdot c) \cdot a + (b + c)^2$$

$$(a + b + c)^2 \text{ collect, } b \rightarrow b^2 + (2 \cdot a + 2 \cdot c) \cdot b + (a + c)^2$$

$$(a + b + c)^2 \text{ collect, } c \rightarrow c^2 + (2 \cdot a + 2 \cdot b) \cdot c + (a + b)^2$$

Команда **Coeffs** використовується для обчислення коефіцієнтів полінома відносно вказаної в команді змінної. Приклад команди **Coeffs** наведений на рис.18.

$$\begin{array}{l}
 (x - b) \cdot (x + b) \text{ coeffs, } x \rightarrow \begin{pmatrix} -b^2 \\ 0 \\ 1 \end{pmatrix} \quad [(4) \cdot x^3 + 3 \cdot x^2] \cdot (x + 1) \text{ coeffs, } x \rightarrow \begin{pmatrix} 0 \\ 0 \\ 3 \\ 7 \\ 4 \end{pmatrix} \\
 (x - b) \cdot (x^2 + 2) \text{ coeffs, } x \rightarrow \begin{pmatrix} -2 \cdot b \\ 2 \\ -b \\ 1 \end{pmatrix} \quad (x + b^2)(b - 1) \text{ coeffs, } b \rightarrow \begin{pmatrix} -x \\ x \\ -1 \\ 1 \end{pmatrix}
 \end{array}$$

Рис.18

Команда **Solve** дозволяє розв'язати рівняння або нерівність з нульовою правою частиною відносно вказаної в цій команді змінної. Приклад використання команди **Solve** наведений на рис.19.

$$\begin{array}{l}
 x^2 + a \cdot x + b \text{ solve, } x \rightarrow \left[\begin{array}{l} \frac{-1}{2} \cdot a + \frac{1}{2} \cdot (a^2 - 4 \cdot b)^{\left(\frac{1}{2}\right)} \\ \frac{-1}{2} \cdot a - \frac{1}{2} \cdot (a^2 - 4 \cdot b)^{\left(\frac{1}{2}\right)} \end{array} \right] \\
 e^x - a \text{ solve, } x \rightarrow \ln(a)
 \end{array}$$

Рис.19

Команда **Solve** дозволяє також розв'язати системи лінійних і нелінійних рівнянь (з нульовою правою частиною). Приклад Розв'язок систем рівнянь наведений на рис.20.

$$\begin{array}{l}
 \left(\begin{array}{l} x^2 + y^2 - 7 \\ 2 \cdot x - y \end{array} \right) \text{ solve, } x, y \rightarrow \left(\begin{array}{l} \frac{1}{5} \cdot \sqrt{35} \quad \frac{2}{5} \cdot \sqrt{35} \\ \frac{-1}{5} \cdot \sqrt{35} \quad \frac{-2}{5} \cdot \sqrt{35} \end{array} \right) \\
 \left(\begin{array}{l} x^2 + y^2 + z^2 - 3 \\ x + y - z - 1 \\ x \cdot y \cdot z - 1 \end{array} \right) \text{ solve, } x, y, z \rightarrow \left(\begin{array}{l} 1 \quad -1 \quad -1 \\ 1 \quad 1 \quad 1 \\ -1 \quad 1 \quad -1 \end{array} \right)
 \end{array}$$

Рис.20

Приклад використання команди *Substitute* наведений на рис.21

$$a \cdot x^2 + b \cdot x + c \text{ substitute ,x = 5} \rightarrow 25 \cdot a + 5 \cdot b + c$$

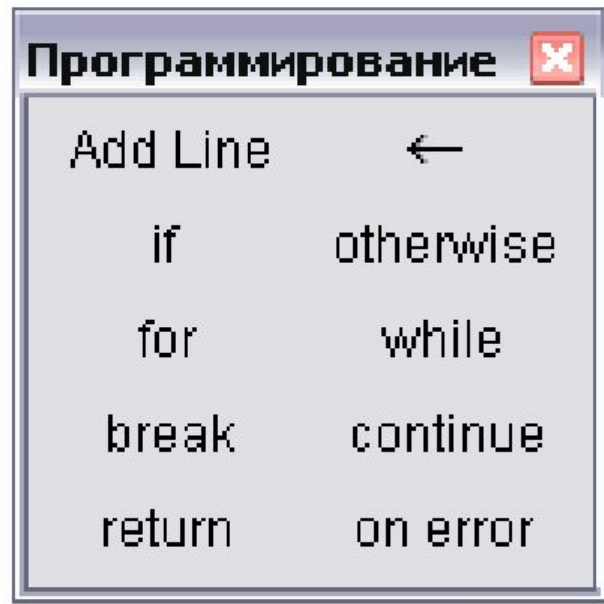
$$a \cdot x^2 + b \cdot x + c \left| \begin{array}{l} \text{substitute ,c = 3} \\ \text{substitute ,x = 5} \end{array} \right. \rightarrow 25 \cdot a + 5 \cdot b + 3$$

$$a \cdot x^2 + b \cdot x + c \left| \begin{array}{l} \text{substitute ,x = 5} \\ \text{substitute ,c = 4} \\ \text{substitute ,b = 11} \end{array} \right. \rightarrow 25 \cdot a + 59$$

$$a \cdot x^2 + b \cdot x + c \left| \begin{array}{l} \text{substitute ,x = 5} \\ \text{substitute ,c = 3} \\ \text{substitute ,b = 4} \\ \text{substitute ,a = 3} \end{array} \right. \rightarrow 98$$

Використання програмних модулів і елементи програмування.

Використання програмних модулів і програмування здійснюються з використанням панелі *Програмування*, яка представлена нижче:.



Робота програмного блоку здійснюється з використанням вертикальної лінії *AddLine*. Усередині програмного блоку можуть виконуватися всі арифметичні операції доступні в *Mathcad*. Особливістю програмного блоку є операція локального привласнення \leftarrow , яка поширює привласнення значення змінної тільки в межах програмного блоку. Приклад такого блоку наведений нижче на рис.22.

Набір програмних елементів для створення програмних модулів містить такі елементи:

- *Add Line* – створює вертикальну лінію, праворуч від якої задається запис програмного блоку;
- < - символ локального привласнення, діє тільки в тілі модуля;
- *if* – умовний оператор;
- *for* – оператор вводу циклу з фіксованим числом повторень;
- *while* – оператор завдання циклу, діючого до тих пір, поки виконується деяка умова;
- *otherwise* – оператор іншого вибору, застосовується з *if*;
- *break* – оператор переривання;
- *continue* – оператор продовження;
- *return* – оператор повернення;
- *on error* – оператор обробки помилок.

Умовний оператор *if* призначений для виконання обчислень залежно від умови:

$$Z(x) := \begin{cases} -1 & \text{if } x < -3 \\ x & \text{if } -3 \leq x < 3 \\ 1 & \text{if } x \geq 3 \end{cases}$$

Це означає, що функція $Z(x)$ приймає значення:

- за першою умовою -1 , якщо $x < -3$;
- за другою умовою x , якщо ;
- за третьою умовою 1 , якщо .

Приклад використання оператора *if* з оператором *otherwise* і без нього наведений нижче:

$$\text{abs}(x) := \begin{cases} s \leftarrow x \\ s \leftarrow -x \text{ if } x < 0 \\ s \end{cases} \quad \text{abs}(-5) = 5 \quad \text{abs}(5) = 5$$

$$\text{abs}(x) := \begin{cases} -x \text{ if } x < 0 \\ x \text{ otherwise} \end{cases} \quad \text{abs}(-5) = 5 \quad \text{abs}(5) = 5$$

$$f(x) := \begin{cases} \text{if } x < 5 \\ \quad \begin{cases} y \leftarrow 2 \\ x \leftarrow x^2 \end{cases} \\ \text{otherwise} \\ \quad \begin{cases} y \leftarrow 4 \\ x \leftarrow x^3 \end{cases} \\ x \cdot y \end{cases} \quad f(8) = 2.048 \times 10^3 \quad f(3) = 18$$

Оператор циклу `for` призначений для створення циклів з фіксованим числом повторень. Шаблон оператора *for* має три мітки:

```
for ■ ∈ ■  
  ■
```

На місці першої мітки вводиться ім'я управляючої змінної; на місці другої мітки вводиться у вигляді ранжируваної змінної початкове і кінцеве значення управляючої змінної (можна також вказати і друге значення управляючої змінної, якщо крок її зміни не рівний одиниці); на місці третьої мітки записується вираз для виконання. Алгоритм роботи оператора циклу *for* такий: управляючій змінній привласнюється перше значення, обчислюється вираз, управляючій змінній привласнюється друге значення, обчислюється вираз і т.д. до перебору всіх значень управляючої змінної.

Приклад використання *оператора циклу for* наведений нижче:

```
f(x) := | s ← 0  
        | for i ∈ 1..5  
        | | s ← s + i  
        | | x ← x · i  
        | x + s
```

$f(5) = 615$

```
sum(n) := | s ← 0  
          | for i ∈ 1..n  
          | s ← s + i
```

$sum(10) = 55$

$sum(20) = 210$

```
prod(n) := | p ← 1  
           | for i ∈ 1..n  
           | p ← p · i  
           | p
```

$prod(3) = 6$

$prod(10) = 3.629 \times 10^6$

Оператор циклу *while* служить для організації циклів, діючих до тих пір, поки виконується умова задана в циклі. Приклад використання *циклу while* наведений нижче:

Приклад обчислення факторіалу

$$\text{Fact}(n) := \left| \begin{array}{l} f \leftarrow 1 \\ \text{while } n \geq 2 \\ \quad \left| \begin{array}{l} f \leftarrow f \cdot n \\ n \leftarrow n - 1 \end{array} \right. \\ f \end{array} \right. \quad \begin{array}{l} \text{Fact}(3) = 6 \\ \text{Fact}(5) = 120 \end{array}$$

Інший приклад обчислення факторіалу

$$f(n) := \left| \begin{array}{l} f \leftarrow n \\ \text{while } 1 \\ \quad \left| \begin{array}{l} f \leftarrow f \cdot (n - 1) \\ n \leftarrow n - 1 \\ \text{break if } n = 1 \end{array} \right. \\ f \end{array} \right. \quad \begin{array}{l} f(3) = 6 \\ f(5) = 120 \end{array}$$

В другому прикладі визначений нескінченний цикл *while*, а примусовий вихід з циклу здійснюється з використанням оператора *break*.

Оператор **Return** використовується для виходу з блоку і передачі значення з будь-якої точки програмного блоку. Приклад використання оператора *Return* наведений нижче:

```
f(x) := | return 1 if x = 0          f(1) = 0.841
        | sin(x) otherwise        f(0) = 1
        | x

g(i) := | return "One" if i = 1    g(1) = "One"
        | return "Two" if i = 2
        | "No value!!!!" otherwise g(3) = "No value!!!!"
```