

Лабораторна робота № 2.

Тема. Операції з числами у системі **MATLAB**. Елементарна математика. Підсистема **Symbolic Math**.

Мета роботи: освоїти роботу з простими функціями, ознайомитись з деякими можливостями та командами **Symbolic Math**.

Теоретичний мінімум

Математичний пакет **MATLAB** включає до себе низьку підсистем, які виконують різні функції. Пакети **MATLAB** є унікальними наборами процедур для розв'язання найрізноманітніших математичних задач.

Symbolic Math Toolbox – це адаптована до мови **MATLAB** версія системи комп'ютерної математики **Maple** фірми **Waterloo Maple, Inc.** Дана підсистема дозволяє використовувати принципи символічної математики, на основі якої можна виконувати такі рутинні операції, як розкриття дужок, перетворення виразів, знаходження коренів рівнянь, похідних функцій та виразів невизначених інтегралів, тощо, і які комп'ютер здійснює самостійно у символічному вигляді без практичного втручання в цей процес користувача.

Вправа 1. Знаходження значення функції при заданих значеннях аргументу. Особливості використання операторів.

Оскільки система **MATLAB** відноситься до матричних систем, передбачається внесення спеціальних коректив при визначенні операторів. У протилежному випадку це приводить до похибок при обчислюванні.

Розглянемо приклад:

```
x =  
    1    2    3    4    5  
>> cos(x)  
ans =  
    0.5403 -0.4161 -0.9900 -0.6536  0.2837
```

Обчислення масиву косинусів пройшло коректно. В той же час при обчисленні функції $\sin(x)/x$ маємо:

```
>> sin(x)/x  
ans =  
   -0.0862
```

Обчислювання масиву значень функції $\sin(x)/x$ дає несподіваний результат – замість масиву з п'ятьма елементами отримано одне значення. Причина в тому, що оператор «/» обчислює відношення двох матриць, векторів або багатовимірних масивів. Якщо вони однієї розмірності, то

результат буде сформований одним числом, що і було здійснене системою у даному випадку. Щоб дійсно отримати вектор значень $\sin(x)/x$, необхідно використати спеціальний оператор поелементного ділення масивів «./». В цьому разі буде отриманий шуканий масив чисел:

```
>> sin(x)./x
ans =
    0.8415    0.4546    0.0470   -0.1892   -0.1918
```

Наведений вище приклад має відношення і до таких операторів, як множення «.*» та зведення в ступінь «.^».

Розглянемо інший приклад. Обчислимо наведену вище функцію, але значенням змінної буде така сукупність: $x = \{0; 1; 2; 3; 4; 5\}$:

```
>> x=0:5;
>> sin(x)./x
Warning: Divide by zero.
(Type "warning off MATLAB:divideByZero" to suppress this
warning.)
ans =
    NaN    0.8415    0.4546    0.0470   -0.1892   -0.1918
```

Зауважимо, що при $x=0$ значення $\sin(x)/x$ дає переборну невизначеність виду $0/0=1$ (Увага: Ділення на нуль). Однак, як і всяка чисельна система, **MATLAB** класифікує спробу розглядання операції ділення на 0 як помилку, та виводить відповідне попередження. Але замість очікуваного чисельного значення виводиться символна константа **NaN**, що позначає невизначеність виду $0/0$ (зауважимо, що невизначеність не є звичайним числом, а символом).

Розглянемо приклади обчислювання функцій при заданому діапазоні значень змінної.

Приклад 1. Обчислити значення функції $y = e^{-2x} \sin x + \log 2x$ при $0.2 \leq x \leq 4$, $\Delta x = 0.5$ ю

Розв'яжемо задачу за допомогою наступних команд:

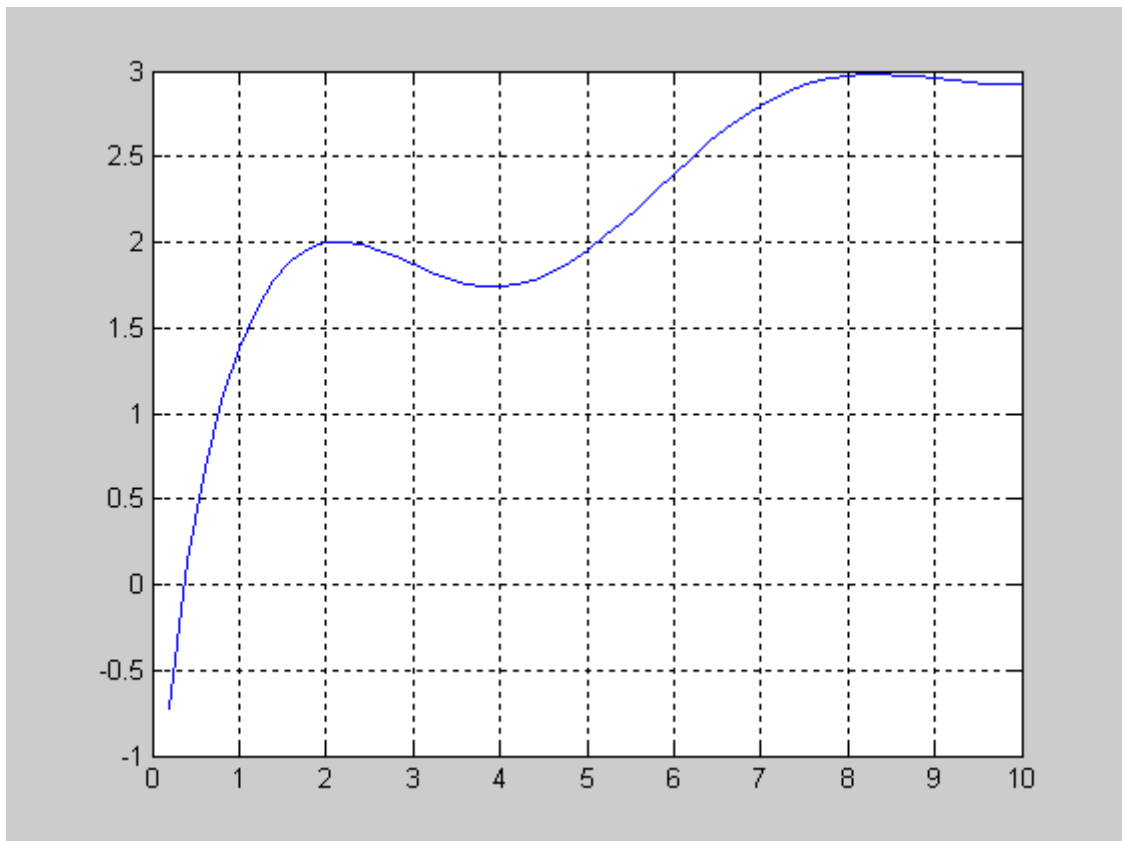
```
>> format long
>> x=0.2:0.2:4;
>> y=exp(-2*x).*sin(x)+log(2*x)
y =
Columns 1 through 4
-0.78311869690974   -0.04816661095642   0.35238860158033
0.61483532616061
Columns 5 through 8
0.80702789462431   0.96002141561858   1.08954467697576
1.20389563290425
Columns 9 through 12
```

```
1.30754298783413  1.40294872443208  1.49153072608432
1.57417480903071
Columns 13 through 16
1.65150242211403  1.72400533826702  1.79210927075520
1.85620099838305
Columns 17 through 20
1.91663799685337  1.97375064653913  2.02784204213370
2.07918766272596
```

Зверніть увагу, що результати обчислювання надані у форматі **long**. Рекомендуємо виконати самостійно таке ж саме обчислювання у форматі **short**.

Для зручності аналізу, а також для показу можливостей системи **MATLAB**, можна побудувати графік обчислюваної функції за допомогою елементарної команди побудови графіків **plot** (детальніше графічні можливості **MATLAB** будуть наведені у лабораторній роботі № 3). В даному прикладі розширимо діапазон значень аргументу x при обчислюванні функції:

```
>> x=0.2:0.2:10;
>> y=exp(-0.2*x).*sin(x)+log(2*x);
>> plot(x,y);grid on
```



Мал. 2.1 – Графік функції $y = e^{-2x} \sin x + \log 2x$ в полі графічного вікна

Вправа 2. Правила використання комплексних чисел. Елементарні операції з комплексними числами та функціями комплексного аргументу.

В системі **MATLAB** дозволяється виконувати обчислення з комплексними числами [33, с.37]. Нагадаємо, що такі числа містять дійсну й мниму частини: $z = x + i \cdot y = \text{Re}(z) + i\text{Im}(z)$. Мнима частина має множник i або j , який називається мнимою одиницею ($i = \sqrt{-1}$):

```
>>b=5*(2.2+3.9i)+0.8  
b =  
11.8000 +19.5000i
```

Якщо коефіцієнтом при мнимій одиниці не є число, то при вводі такого комплексного числа необхідно користуватись знаком множення. Наприклад, замість виразу $a + ib$ треба вводити $a + i*b$, щоб програма не видала повідомлення про помилку.

Найпростіші операції з комплексними числами: операція додавання «+», операція віднімання «-», операція множення «*», операція ділення зліва направо «/», операція ділення справа наліво «\», операція піднесення до степеня «^». Ці операції здійснюються за допомогою арифметичних знаків, що наводяться у дужках, наприклад:

```
>> format long;  
>> x=2+3i;  
>> y=-2+5i;  
>> disp(x+y)  
0 +8.000000000000000e+000i  
>> disp(x-y)  
4.000000000000000e+000 -2.000000000000000e+000i  
>> disp(x/y)  
3.793103448275862e-001 -5.517241379310345e-001i  
>> disp(x*y)  
-1.900000000000000e+001 +4.000000000000000e+000i  
>> disp(x\y)  
8.461538461538463e-001 +1.230769230769231e+000i  
>> disp(x^y)  
-1.482729564776616e-004 -5.450608337156012e-004i
```

У даному фрагменті використовується функція **disp** (від слова «дисплей»), що дозволяє виводити в командне вікно результати обчислень або будь-який текст. При цьому результат обчислень (число) виводиться без вказівки імені змінної чи імені **ans**, яким звичайно повинен присвоюватись результат.

При використанні комплексних аргументів є можливість обчислити всі елементарні математичні функції, що задаються в системі **MATLAB**. Наприклад:

```

>> x=2+3i; y=-1+5i;
>> disp (sqrt(x))
1.674149228035540e+000 +8.959774761298381e-001i
>> disp (sin(x))
9.154499146911430e+000 -4.168906959966565e+000i
>> disp (exp(y))
1.043534862696817e-001 -3.527685262888061e-001i
>> disp (abs(y))
5.099019513592785e+000

```

У системі **MATLAB** існують кілька додаткових функцій, розрахованих лише на комплексний аргумент. За допомогою наведених вище команд можна виконати наступні операції:

```

>> x=2+3i; y=-1+5i;
>> disp (real(y))
-1
>> disp (imag(x))
3
>> disp (angle(x))
9.827937232473291e-001
>> disp (conj(y))
-1.000000000000000e+000 -5.000000000000000e+000i

```

Звернемо увагу на існування спеціальної функції **cxpair(V)**. Дана функція здійснює сортування заданого вектора (**V**) з комплексними елементами таким чином, що комплексно-спряжені пари цих елементів розташовуються у вихідному векторі у порядку зростання їх дійсних частин. При цьому елемент з від'ємною уявною частиною завжди розташовується першим.

Вправа 3. Підсистема **Symbolic Math Toolbox**. Команди та функції символічних об'єктів.

Підсистема **Symbolic Math Toolbox** – пакет прикладних програм, що дають системі **MATLAB** принципово нові можливості, — можливості розв'язку задач в символічному (аналітичному) вигляді, що включають реалізацію точної арифметики довільної розрядності. Пакет базується на застосуванні ядра символічної математики однієї з наймогутніших систем комп'ютерної алгебри — **Maple V R4**. **Symbolic Math** забезпечує виконання символічного диференціювання і інтегрування, обчислення сум і добутків, розкладання в ряди Тейлора і Маклорена, операції з поліномами, обчислення коренів поліномів, розв'язок в аналітичному виді нелінійних рівнянь, різні символічні перетворення, підстановки, тощо.

Пакет дозволяє готувати процедури з синтаксисом мови програмування системи **Maple V R4** і встановлювати їх в системі **MATLAB**. Але по

можливостях символної математики пакет сильно поступається спеціалізованим системам комп'ютерної алгебри, таким як нові версії **Maple** і **Mathematica**.

Для одержання переліку доступних в системі **MATLAB** команд системи символної математики **Maple** можна звернутися до довідкової системи та набрати назву команди у рядку вводу команди:

```
>> help symbolic
```

У процесі виконання команд **Symbolic Math Toolbox** результати надаються змінним **MATLAB** і можуть бути використані у інших режимах обчислювання, візуалізації і т.д. [20, с.453]. Для одержання довідки про математичні команди із символного пакету **Symbolic Math Toolbox** варто попереду імені команди записати префікс **sym**:

```
>> help sym/diff
```

DIFF Differentiate.

DIFF(S) differentiates a symbolic expression S with respect to its

free variable as determined by FINDSYM.

DIFF(S,'v') or DIFF(S,sym('v')) differentiates S with respect to v.

DIFF(S,n), for a positive integer n, differentiates S n times.

DIFF(S,'v',n) and DIFF(S,n,'v') are also acceptable.

Examples;

```
x = sym('x');
```

```
t = sym('t');
```

```
diff(sin(x^2)) is 2*cos(x^2)*x
```

```
diff(t^6,6) is 720.
```

See also INT, JACOBIAN, FINDSYM.

Існує інший варіант отримання довідки про команду з системи **Maple**, наприклад, у формі наступного запиту:

```
>> mhelp diff
```

Рекомендуємо виконати цей запис та ознайомитися з наданою інформацією.

Для роботи з командами ядра системи **Maple** у **MATLAB** визначено новий тип об'єкту **sym** – символний об'єкт (**symbolic object**). Для проведення аналітичних операцій інтегрування, диференціювання та інших потрібно, щоб відповідні аргументи функцій були попередньо оголошені. Група символних змінних може бути створена за допомогою описувача **syms**. Наприклад:

```
>> syms s1 s2
```

Для введення однієї символної змінної можна застосувати команду **sym**:

```
>> s3=sym('s3')
s3 =
s3
```

Нова символна змінна, яка виражається через попередньо задані змінні, може бути визначена наступним чином:

```
>> s4=(cos(s1)*s2+sqrt(5)*s3)
s4 =
cos(s1)*s2+5^(1/2)*s3
```

Для представлення аналітичного виразу в більш звичайній для розуміння формі використовується команда **pretty**:

```
>> pretty(s4)
              1/2
cos(s1) s2 + 5  s3
```

Виявити символні змінні у заданому виразі дозволяє команда **findsym**:

```
>> findsym(s4)
ans =
s1, s2, s3
```

Команда **sym** використовується також для введення абстрактної функції:

```
>> f=sym('f(x)')
f =
f(x)
```

Вправа 4. Аналітичні перетворення за допомогою команд підпакету **Symbolic Math Toolbox**.

Робота у системі **MATLAB** відрізняється від прийнятого у системі **Maple** стилю. Наприклад, треба визначити модуль комплексного виразу:

```
>> syms x y real; az=abs(x+i*y)
(x^2+y^2)^(1/2)
```

Надамо змінній **y** числове значення і застосуємо змінну **az**, яка містить змінну **y**:

```
az =
>> y=1;az
az =
(x^2+y^2)^(1/2)
```

Автоматичної підстановки значення змінної не відбулося. У системі **MATLAB** треба звернутися до спеціальної команди **subs**, щоб привласнити змінним, що приймають участь у виразі **az**, ті значення, які вони отримують в процесі розрахунків:

```
>> subs(az)
ans =
(x^2+1)^(1/2)
```

Команда **subs** має наступний формат:

```
>> subs(s,old,new),
```

де **s** – вираз, у якому здійснюється підстановка нових виразів (**new**) замість існуючих (**old**), де **new**, **old** – символічні змінні, рядки, визначені за допомогою квадратних дужок. Наприклад:

```
>> syms x y
>> z=x^2+y^2
z =
x^2+y^2
>> d=subs(z,[x,y],[cos(x),sin(x)])
d =
cos(x)^2+sin(x)^2
```

Автоматичного спрощення виразу не відбувається. Щоб реалізувати процес спрощення, використовуємо команду **simplify**:

```
>> simplify(d)
ans =
1
```

Крім потужної багатоцільової команди **simplify** у системі **MATLAB** заслуговує уваги команда **simple**, яка дає можливість переглянути результати застосування різних операцій.

Розглянемо результат дії команди **simple** у випадку елементарного підведення у квадрат змінної **d**:

```
>> simple(d^2)
simplify:
1
radsimp:
(cos(x)^2+sin(x)^2)^2
combine(trig):
1
factor:
(cos(x)^2+sin(x)^2)^2
expand:
cos(x)^4+2*cos(x)^2*sin(x)^2+sin(x)^4
combine:
1
convert(exp):
((1/2*exp(i*x)+1/2/exp(i*x))^2-1/4*(exp(i*x)-
1/exp(i*x))^2)^2
convert(sincos):
(cos(x)^2+sin(x)^2)^2
convert(tan):
```



```

((1-
tan(1/2*x)^2)^2/(1+tan(1/2*x)^2)^2+4*tan(1/2*x)^2/(1+tan(1/2
*x)^2)^2)^2
collect(x):
(cos(x)^2+sin(x)^2)^2
ans =
1

```

Наведемо назви команд для аналітичних операцій, які застосовуються у випадку перетворення виразів:

- **subs** – операція підстановки;
- **subexpr** – операція запису з підстановками та використанням проміжних величин;
- **simplify** – операція спрощення виразу;
- **simple** – операція спрощення виразу з перерахуванням варіантів;
- **expand** – операція розкриття дужок;
- **factor** – операція розкладання виразу на множники;
- **collect** – операція перетворення вираження в поліном з виділенням коефіцієнтів при ступенях заданих змінних;
- **numden** – операція приведення до раціональної форми;