**Національний лісотехнічний університет України**

**Кафедра інформаційних технологій**

**Думанський О.І., Думанський Н.О.,**

**Шиманський В.М.**

**Чисельна реалізація задач операційного числення у середовищі Matlab**

**ЛАБОРАТОРНИЙ ПРАКТИКУМ**

**ЛЬВІВ – 2016**

**Думанський О. І. , Думанський Н. О. , Шиманський В.М.**

**Чисельна реалізація задач операційного числення у середовищі Matlab**

**Лабораторний практикум**

Лабораторний практикум укладений для виконання лабораторних робіт з дисципліни **«Операційне числення»**. У практикумі розроблені лабораторні роботи, для кожної з яких наведено у скороченому вигляді відповідний теоретичний матеріал, лістинги програмних кодів у середовищі Matlab та приклади розв’язку поставленого практичного завдання або методичні вказівки та алгоритми шляху реалізації завдання. До кожної лабораторної роботи побудовані індивідуальні завдання та перелік контрольних питань.

Цей практикум може бути використаний для студентів різних форм навчання, для яких читається дисципліна **«Операційне числення»**.

Матеріали лабораторного практикуму обговорені та затверджені до друку на засіданні кафедри ІТ, протокол №\_***6***\_ від\_\_***12 грудня 2016 р.***\_

**Рецензенти**: Поберейко Б. П. доктор техн. наук, професор зав. кафедрою автоматизації та комп’ютерних технологій Національного лісотехнічного університету України,

Флячок В. М. кандидат фіз. – матем. наук, доцент кафедри прикладної

математики і комп’ютерних інформаційних систем Академії друкарства.

**Укладачі:** Думанський О. І. кандидат фіз. – матем. наук, доцент кафедри інформаційних технологій Національного лісотехнічного університету України,

Думанський  Н. О. старший викладач кафедри соціальних комунікацій та інформаційної діяльності Національного університету «Львівська політехніка»,

Шиманський В.М. старший викладач кафедри інформаційних технологій Національного лісотехнічного університету України.

**ВСТУП**

В останні роки наукове програмування зазнало серйозної трансформації: розвиваються інтегровані середовища, що ґрунтуються на спеціалізованих алгоритмічних мовах, зростає застосування універсальних математичних систем типу MATLAB, MATHEMATICA, MATCAD та ін. Ці системи мають дружній інтерфейс, потужні графічні засоби, власні мови програмування, реалізують ряд стандартних і спеціальних математичних операцій. Все це надає широкі можливості для ефективної роботи спеціалістів різних галузей, про що говорить активне застосування математичних пакетів у наукових дослідженнях і викладацькій діяльності. Математичні пакети ­ – гарний вибір для проведення будь-якого дослідження, де вимагається математика, – від курсової роботи до наукового пошуку.

Навчальне видання призначено для ознайомлення студентів з матема­тичним пакетом MATLAB, який вивчається у відповідності з програмою дисципліни “Прикладні програми”, що викладається на механіко-матема­тичному факультеті. Серед інших пакетів обрано MATLAB з причини поступового розвитку універси­тетського процесу навчання. MATLAB дозволяє студенту природно перейти у своєму навчанні від простих обчислень до можливостей програмування високого рівня через використання потужних обчислювальних і графічних модулів. Останнє означає, що студенти можуть займатися програмуванням в MATLAB навіть без попереднього вивчення його основ. Проста вбудована мова програмування дозволяє легко створювати власні алгоритми.

В MATLAB реалізовані класичні чисельні алгоритми розв’язання рівнянь, задач лінійної алгебри, знаходження значень визначених інтегралів, апроксимації, розв’язання окремих рівнянь і систем диференціальних рівнянь. Для застосування більшості можливостей пакета достатньо базових знань з чисельних методів. Розв’язання спеціальних задач, зрозуміло, вимагає відповідної теоретичної підготовки.

За змістом видання поділено на теми, які містять основні відо­мості з операційного середовища MATLAB, елементів мови, матричних обчислень, графіки, чисельного аналізу та конкретних прикладів опису комп’ютерних технологій розв’язання прикладних задач математичної фізики за допомогою MATLAB.

Навчальне видання може бути корисне студентам університетів при проведенні вправ на персональному комп’ютері, наприклад, з обчислювальної математики, інформатики та інших предметів, та контрольних і лабораторних робіт із спеціальних дисциплін, що вимагають елементів наукових досліджень. Викладений матеріал доцільно також використовувати для обробки результатів математичних досліджень при курсовому і дипломному проектуванні.

**Розділ І. Загальна характеристика операційного середовища системи MATLAB**

**§1. Призначення і сфера застосування математичного пакета MATLAB.**

**Загальна характеристика середовища.** Для

***Лабораторна робота №1***

***Тема роботи: операції над комплексними числами***

***Мета роботи***:освоїти апарат дій з комплексними числами та методику їх опрацювання у середовищі MATLAB

***Теоретичні засади***

Комплексні змінні мають тип double і не потребують ніякого попереднього опису. Для запису уявної одиниці зарезервовані букви i або j. У випадку, коли коефіцієнтом перед уявною одиницею є не число, а змінна, між ними потрібно використовувати знак множення.

Комплексні числа можна записувати таким чином:

» 2+3i; -6.789+0.834e-2\*i; 4-2j; x+y\*i;

Майже всі елементарні функції допускають обчислення з комплексними аргументами. Обчислити вираз:

» res=sin(2+3i)\*atan(4i)/(1-6i)

Отримаємо результат:

-1.8009 - 1.9190i

Для роботи з комплексними числами призначені такі функції:

***real*** - дійсна частина комплексного числа;

***imag*** - уявна частина комплексного числа;

***abs*** - аболютне значення комплексного числа;

***angle*** - аргумент комплексного числа;

***conj*** - комплексно спряжене число).

Окрім вищенаведених операцій можна виконувати наступні дії над комплексними числами:

* додавання;
* віднімання;
* множення;
* ділення;
* піднесення до степеня;
* добування коренів, але при цьому потрібно задати комплексне число у тригонометричній формі згідно формули Муавра : , де k=0, 1, …, n-1, n – показник кореня.

Приклади:

1. Обчислення модуля і аргументу комплесного числа.

Якщо комплесне число подано у вигляді , то його модуль *r* та аргумент *argz* визначаємо згідно формул 

*% модуль і аргумент %*

*>> z=2+3i;*

*>> r=abs(z)*

*r = 3.6056*

*argf=atan(3/2)*

*argf = 0.9828*

1. Операції над комплексними числами:

>> z1=2+3i; z2=4-5i;

>> % додавання%

>> w1=z1+z2

w1 = 6.0000 - 2.0000i

>> % віднімання%

>> w2=z1-z2

w2 = -2.0000 + 8.0000i

>> % множення%

>> w3=z1\*z2

w3 = 23.0000 + 2.0000i

>> % ділення%

>> w4=z1/z2

w4 = -0.1707 + 0.5366i

>> % піднесення до степеня %

>> w5=z1^3

w5 = -46.0000 + 9.0000i

Операцію піднесення до степеня перевірити, подавши комплексне число у тригонометричній формі згідно формули



Операцію добування кореня здійснити, за поданим комплексним числом у тригонометричній формі , згідно формули : , де k=0, 1, …, n-1, n – показник кореня, наступним чином:

**>>** *z1=2+3i;*

*>> k=0:1:3; f=atan(3/2)*

*f =0.9828*

>> z=abs(z1)^(1/4)\*(cos((f+2\*k.\*pi)/4)+i\*sin((f+2\*k.\*pi)/4))

При k=0, z1 =1.3366 + 0.3352i;

k=1, z2= -0.3352 + 1.3366i;

k=2, z3= -1.3366 - 0.3352i;

k=3, z4= 0.3352 - 1.3366i.

Таким чином, маємо чотири корені комплексно числа.

*Індивідуальні завдання*

Завдання 1. Задані комплексні числа 

Потрібно**:** 1. Знайти модуль і головне значення аргументу комплексних чисел *z1* та *z2*.

1. Провести всі операції над поданими числами, за винятком добування коренів.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Варіант | Z1 | Z2 | Варіант | Z1 | Z2 |
| 1 | z1= 1+2i | z2=-1+i. | 7 | z1=1+ | z2=-1- |
| 2 | z1= 1-i | z2= -1-i | 8 | z1=--i | z2=i |
| 3 | z1= -8+i8 | z2=-3i | 9 | z1= | z2= |
| 4 | z1=-i | z2=- | 10 | z2= | z1= |
| 5 | z1=i | z2=8+i8 | 11 |  |  |
| 6 | z1= +i | z2=-+i | 12 | z1= | z2=- |

Завдання 2. Задане комплексне число . Знайти всі значення .

1.  6.  11. 

2.  7.  12. 

3.  8.  13. 

4.  9.  14. 

5.  10.  15. 

***Контрольні запитання***

1. Як ввести значення комплексного числа, і у якому вигляді воно виведеться на екран?

2. Як у середовищі MatLAB забезпечити дії над комплексними ?

3. Які функції обробки комплексних чисел передбачені у середовищі

MatLAB?

***Лабораторна робота №2***

***Розв’язування системи рівнянь з комплексними коефіцієнтами***.

***Мета роботи***: освоїти методику розв’язування системи лінійних алгебраїчних рівнянь, якщо елементи матриці системи та вільні члени є комплексними числами

***Теоретичні засади***

Часто при розрахунку електричного кола приходиться мати справу з системою лінійних алгебраїчних рівнянь, якщо елементи матриці системи та вільні члени є комплексними числами. Шлях розв’язування такої системи є аналогічний розвязку таких рівнянь, коли елементи матриці системи та вільні члени є дійсними числами. В даному випадку скористаємось матиричним методом, оскільки він є найпростішим та швидким.

Якщо маємо систему рівнянь , де *А*- матриця системи, елементами котрої є коефіцієнти системи, *Х* – матриця- стовпець невідомих і *В* – матриця- стовпець вільних членів, то розв’язок системи шукаємо у вигляді

, *А-1* – обернена матриця матриці *А.*

***Приклад***

Розв’язати систему рівнянь



Розв’язок системи у середовищі Matlab:

>>A=[2+3i 4 3-i; 1-i 2i 5; 3-2i 6+3i 6]

A = 2.0000 + 3.0000i 4.0000 3.0000 - 1.0000i

1.0000 - 1.0000i 0 + 2.0000i 5.0000

3.0000 - 2.0000i 6.0000 + 3.0000i 6.0000

>> B=[-2+3i -5i 7-i]

B = -2.0000 + 3.0000i 0 - 5.0000i 7.0000 - 1.0000i

>> B1=B'

B1 = -2.0000 - 3.0000i

0 + 5.0000i

7.0000 + 1.0000i

>> X=inv(A)\*B1

X = -0.4669 + 1.1375i

1.0212 - 1.3390i

-0.6697 + 0.2706i

>> % Перевірка

>> C=A\*X

C = -2.0000 - 3.0000i

0.0000 + 5.0000i

7.0000 + 1.0000i

***Інвидуальні завдання***

Завдання 3. Знайти корені, поданої у комплексному вигляді системи рівнянь:

1. 

2. 

3. 

де m - передостання цифра, а n - остання цифра студентського квитка .

***Контрольні запитання і вправи***

***Лабораторна робота 3***

***Тема роботи:* дії з комплексними числами, поданими у відповідних формах**

***Мета роботи:*** *виконати вказані дії над комплексними числами:*

*а*) число z1, подане в алгебраїчній (експоненціальній) формі, переведіть

в експоненціальну (алгебраїчну), перевірте і запишіть результат;

б) число z2, подане в експоненціальній (алгебрїчній) формі, переведіть

в експоненціальну (алгебраїчну), перевірте і запишіть результат;

в) обчисліть поданий вираз і запишіть результат в експоненціальній

формі, при цьому аргумент результату повинен бути у межах між (-π) і π

***Теоретичні засади***

У деяких випадках потрібно виконати операції над комплексними числами поданими у різних формах їх подання, тобто у алгебричній, показниковій або тригонометричній. Для подання, комплексного числа поданого в алгебраїчній формі, у тригонометричній та показниковій, потрібно знайти його модуль та аргумент, а дальше згідно відповідних формул записуємо це число у відповідній формі. Для проведення різного типу обчислень над комплексними числами, поданими в алгебраїчній і показниковій формі у середовищі Matlab немає потреби зводити їх до однієї форми.

Якщо є потреба у зведенні комплексного числа до показникової або тригонометричної форми, потрібно обчислити його модуль та аргумент, а дальше записуємо число у потрібній формі. У середовищі Matlab є відповідні функції, згідно яких встановлюємо модуль та аргумент поданого комплексного числа: ***abs*** - аболютне значення комплексного числа;

***angle*** - аргумент комплексного числа.

Для виконання цієї лабораторної роботи наведено взірець її виконання.

Взірець лістингу програмного коду виконання лабораторної роботи

% Вводимо подані комплексні числа і після їх введення

% здійснюємо їх перетворення у відповідну форму згідно завдання

>> z1=6.7\*exp(-pi/12\*i) %перетворюємо в алгебричну форму

z1 = 6.4717 - 1.7341i

>> % Перевірка

>> r1=abs(z1)

r1 = 6.7000

>> f1=angle(z1)

f1= -0.2618

>> f1=atan(-pi/12)

f1= -0.2618

>> z2=5.3-6.2i; % Перетворюємо у показникову форму

>> % Знаходимо модуль та аргумент поданого числа

z2=5.3-6.2i;

>> r2=abs(z2)

r2 = 8.1566

>> f2=angle(z2)

f2 =-0.8635

>>% Перевірка

>> f2=atan(-6.2/5.3)

f2 = -0.8635

>> %Запис комплексного числа у показниковій формі

>> % z2=8.1566\*exp(-0.8635\*i)

% Обчислення поданого виразу

% Введення поданих комплексних чисел

>> z1=5-8i; z2=6\*exp(pi/12\*i);

>> z3=7\*exp(-pi/6\*i); z4=8+7i;

% Обчислення поданого виразу

>> z=z1^2\*(z2-z3)/z4

z = 18.4683 -38.1265i

% Знаходимо його модуль та аргумент

>> r=abs(z)

r =42.3640

>> f=angle(z)

f =-1.1197

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Варіант | Комплексне число  z1 z2 z 3 z4 | | | | Вираз |
| 1 | 2+3i |  |  |  |  |
| 2 | 0,8− 2*i* |  |  | −5+*i* |  |
| 3 | −0,7+4*i* |  | 3+4*i* |  |  |
| 4 | −3−2*i* |  | 1,2 + 3*i* |  |  |
| 5 |  | − 0,7+ 4*i* |  | −8−3*i* |  |
| 6 |  | −3−2*i* |  | +*i* |  |
| 7 |  | 0,8− 2*i* |  | −*i* |  |
| 8 | 4+3*i* | 3,21*e1.5i* |  |  |  |
| 9 | 1 + *i*π/ 2 |  | 0,8− 2*i* |  |  |
| 10 |  |  |  | −3 - 2*i* |  |
| 11 | 0,187− 3,94*i* |  | − 0,7+ 4*i* |  |  |
| 12 | −1+*i* |  |  | 4+3*i* |  |
| 13 | −−4*i* |  | −3−2*i* |  |  |
| 14 |  | 0,18 − 3,9*i* |  | 0,8− 2*i* |  |
| 15 |  | −1+*i* | -0,7+ 4*i* |  |  |
| 16 |  | −−4*i* | 4+3*i* |  |  |
| 17 |  | −*i* |  | − 0,7+ 4*i* |  |
| 18 |  | 1+ *i*π/ 2 | +*i* |  |  |
| 19 | 2,7+ 0,8*i* |  |  | −−*i* |  |
| 20 | −0,8+2,7*i* | - |  | 3,1− 2,1*i* |  |

***Контрольні запитання і вправи***

***Лабораторна робота №4***

***Тема роботи: елементарні операції з функцією комплексної змінної***

***Мета роботи: з***находження модуля і головного значення аргументу функції комплексної змінної та обчислення функції при відповідних комплексних числових значеннях комплексної змінної z=a+ib .

***Теоретичні засади та лістинги програмних кодів***

При поданій функції комплексної змінної у деяких випадках потрібно встановити значення її модуля та аргументу по поданому числовому значенні комплексної змінної. Безпосередній шлях знаходження цих величин можна вибрати наступним чином: підставляємо числове значення незалежної змінної ***z=a+ib****,* дальше виділяємо дійсну та уявну частини функції, а тоді згідно формул знаходження модуля та головного значення аргументу визначаємо вказані величини.

Проте цю операцію за допомогою середовища Matlab можна зробити простіше та ефективніше згідно функцій, які подані вище у першій лабораторній роботі, а саме :

***abs*** - аболютне значення комплексного числа;

***angle*** - аргумент комплексного числа.

Для прикладу розглянемо знаходження модуля і головного значення аргументу функції комплексної змінної w=ctgz+2z при z=.

Побудуємо лістинг програмного коду завдання.

>>% Знаходження модуля та аргументу функції w=ctgz+2z

>>% при z=

>>% Подаємо значення змінної z

>> z=sqrt(8)+i\*sqrt(24);

>>% Обчислюємо модуль функції r

>> r=abs(cot(z)+2\*z)

r= 10.4595

>>% Обчислюємо головне значення аргументу функції

>> argz=angle(cot(z)+2\*z)

argz=0.9994

Якщо потрібно безпосередньо обчислити значення функції при поданому числовому значенні комплексної змінної, то цей процес теж відповідно простим та зручним. Наприклад потрібно обчислити значення функції

w1= , w2=  , w3= .

>> % Лістинг програмного коду

>> z1=-3-i\*3\*sqrt(3);

>> w1=z1^(1/3)

w1=1.3920-1.1680i

>> z2=1+4i;

>> w2=cos(z2)

>> w2=14.7547-22.9537i

>> z3=-4i;

>> w3=log(z3)

w3=1.3863-1.5708i

***Індивідуальні завдання***

***Завдання 1***. Знайти модуль і головне значення аргументу функції w=f)z) при трьох значеннях аргументу *z=a+ib.*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Варіант | | | Функція | | Z1 | | Z2 | | Z3 | |
| 1 | | | w=cosz+2z2 | | z1= 1+2i | | z2=-1+i | | z3=1+i | |
| 2 | | | w=e2z+2z2 | | z1= 1-i | | z2= -1-i | | z3=-i | |
| 3 | w=3-2z+3z2 | | z1= +i | | z2=-+i | | z3=- | |
| 4 | w=cos5z+ | | z1= | | z2=i | |  | |
| 5 | w=z+2z2 | | z1= | | z2= | |  | |
| 6 |  | | z1= | | z2= | |  | |
| 7 |  | | z1= | | z2=1- | |  | |
| 8 | *w = (i+1)z+i* | | z1=, | | z2= | |  | |
| 9 |  | | z1=i | | z2=-3i | |  | |
| 10 | w=-2z-tgz | | z1=-i | | z2=- | |  | |
| 11 |  | |  | | z2= | |  | |
| 12 | w=ctgz+2z | | z1= | | z2= | |  | |

***Завдання 2.*** Обчислити значення функції при заданому значенні її аргументу.

|  |  |  |  |
| --- | --- | --- | --- |
| Варіант | Функція | Функція | Функція |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |
| 11 |  |  |  |
| 12 |  |  |  |

***Контрольні запитання та вправи***

1. По вектору *Z*=[1-2*i*, 1+3*i*, 1+2*i*, 2, 5*i*, 0] створити матрицю: перший рядок – це модуль відповідних компонент вектора *Z*, другий – аргумент, третій – дійсна частина, четвертий – уявна частина.
2. Для вектора *Z*  з завдання 2 знайти скалярний добуток *Z*  на *Z*\**.*
3. Для натурального числа *N* і комплексного числа **** знайти розв’язок рівняння 

***Лабораторна робота №5***

***Тема роботи: обчислення похідних функції комплексної змінної***

***Мета роботи:*** освоїти процес знаходження похідних функції комплексної змінної 

***Теоретичні засади***

В MATLAB похідна знаходиться за допомогою функції:

**diff (w, z, n)**

w – функція, яку потрібно продиференціювати;

z – аргумент функції (змінна диференціювання);

n – порядок похідної (по замовчуванню n = 1).

Щоб знайти похідну, потрібно:

* Визначити символьні змінні з допомогою функції **syms** ;
* Ввести функцію, яку потрібно продиференціювати **f**;
* Ввести функцію диференціювання **diff (w, z, n)**;
* Отримати результат після натиснення ***Enter***.

***Приклад 2***: нехай потрібно знайти першу і третю похідну функції xcosx.

>> syms z;

>> w= z \* cos (z);

>> diff (w, z)

ans = -sin (z) \* z + cos (z)

>> diff (w, z, 3)

ans = sin (z) \* z – 3 cos (z)

***Індивідуальні завдання***

***Завдання.***  Обчислити перші та треті похідні поданих функцій

|  |  |  |  |
| --- | --- | --- | --- |
| Варіант | Функція | Функція | Функція |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |

***Лабораторна робота №6***

***Тема роботи: обчислення інтегралів функції комплексної змінної (ФКЗ)***

***Мета роботи:*** освоїти процес знаходження інтегралів від аналітичних функцій комплексної змінної (ФКЗ)

***Теоретичні засади***

Система Matlab дає можливість одержати у числовому або функціональному вигляді значення інтегралу аналітичної функції комплексної змінної (ФКЗ). Оскільки, при аналітичності ФКЗ процес інтегралу аналогічний як і для функції дійсної змінної, то всі прийоми обчислення є ідентичними. Тобто, для реалізації цієї символьних операцій у системі, потрібно попередньо описати символьні змінні згідно команди **syms** з переліком змінних, через які виражається функція, інтеграл якої шукаємо.

*Розглянемо обчислення визначених інтегралів* *ФКЗ*

Для реалізації цієї операцій маємо функції:

***int(f,a,b)*** – обчислює визначений інтеграл від функції ***f*** з межами інтегрування ***a*** та ***b***, причому ці межі можуть бути як числовими, так і символьними. І , якщо аргументом підінтегральної функції є не х, а інша змінна, наприклад, z***,*** то запис функції інтегрування має вигляд ***int (f,z ,a/b)***.

Наведемо приклади побудови програмних кодів для знаходження поданих інтегралів.

а)Обчислити інтеграл  , де *L*- лінія, що з’єднує точки *z1=I, z2=1+2i.*

Лістинг програмного коду:

>> % Обчислення поданого інтегралу

>> syms z;

>> f=z^2\*exp(z);

>> F=int(f, 1,1+2\*i)

F =-3\*exp(1+2i) – exp(1)

>> Для отримання числового значення

>>I= eval(F)

I=0.6753-7.4153i

б) Обчислити інтеграл  , де *L*- лінія, що з’єднує точки *z1=i, z2=3i.*

>> syms z;

>>f=1/(sin(z)^2\*(1-cos(z)));

>>F=int(f, i,3\*i)

F =-1/12\*i\*(-3\*tanh(1/2)^3\*tanh(3/2)^4+tanh(1/2)^3- 6\*tanh(1/2)^3\*tanh(3/2)^2+3\*tanh(3/2)^3\*tanh(1/2)^4-tanh(3/2)^3+6\*tanh(3/2)^3\*tanh(1/2)^2)/tanh(3/2)^3/tanh(1/2)^3

>>I=eval(F)

I = 0 + 0.3132i

***Індивідуальні завдання***

***Завдання 3***. Знайти невизначений та обчислити визначений інтеграли

|  |  |  |  |
| --- | --- | --- | --- |
| № ***вар.*** | ***Інтеграли*** | № ***вар.*** | ***Інтеграли*** |
| 1 | (z1=1, z2=2i) *€ L* | 6 | ; |
| 2 | (z1=-2, z2=0) *€ L* | 7 |  |
| 3 | (z1=1, z2=3) *€ L* | 8 |  |
| 4 | (z1=1, z2=e2) *€ L* | 9 |  |
| 5 | (z1=0, z2=1) *€ L* | 10 |  |

***Контрольні завдання та вправи***

***Лабораторна робота №7***

***Тема роботи: пряме і обернене перетворення Лапласа***

***Мета роботи:*** освоїти апарат перетворення Лапласа та шляхи знаходження прямого та оберненого перетворення Лапласа.

***Теоретичні основи***

1. **Пряме перетворення Лапласа**

Метод перетворення Лапласа полягає в тому, що тут вивченню підлягає не функція ***f*(t)**, яка називається **оригіналом**, а її **зображення** **F(p)**. Це зображення здійснюється за допомогою множення оригіналу на деяку екпоненціальну функцію і цей добуток інтегрується в межах від 0 до ∞:



де  - оригінал функції;  - її зображення.

Перетворення Лапласа є інтегральним перетворенням, яке позначається символом: .

В системі MATLAB перетворення Лапласа оригіналу функції  здійснюється за допомогою функції:

**Laplace (f(t))**

***Приклад 1***. Згідно інтегрального перетворення Лапласа знайти зображення F(p) функції оригіналу *f*(t) = .

**>> syms t**

**>> q = exp(-2\*t);**

**>> Laplace (q)**

**ans = 1/s+2**

Отже, можна записати:

.

**2. Обернене перетворення Лапласа**

В системі MATLAB обернене перетворення Лапласа здійснюється за допомогою функції:

**iLaplace (L, t)**

де L(s) – пряме перетворення Лапласа, t – аргумент шуканої функції оригіналу *f*(t).

***Приклад 2***. Знайти оригінал функції *f*(t), якщо зображення її .

**>> syms a b s**

**>> L = (a+b\*s)/s^2;**

**>> iLaplace (L, t)**

**ans = at + b**

***Індивідуальні завдання***

***Завдання 1***. Згідно інтегрального перетворення Лапласа знайти зображення *F(p)* функції оригіналу *f*(t)

|  |  |  |  |
| --- | --- | --- | --- |
| Варіант | Функція | Функція | Функція |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |

***Завдання 2***. Згідно оберненого перетворення Лапласа знайти оригінал функції *f*(t), якщо задано її зображення.

|  |  |  |  |
| --- | --- | --- | --- |
| Варіант | Зображення | Зображення | Зображення |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |

***Контрольні запитання та вправи***

***Лабораторна робота №8***

***Тема роботи: пряме та обернене перетворення Фур’є***

***Мета роботи:*** освоїти апарат та шляхи знаходження прямого та оберненого перетворення Фур’є.

***Теоретичні основи***

1. **Пряме перетворення Фур’є**

Метод перетворення Фур’є полягає в тому, що тут вивченню підлягає не функція ***f*(t)**, яка називається ***оригіналом****,* а *її* ***зображення*** **Х(ω).** Це зображення

здійснюється за допомогою множення оригіналу на деяку екпоненціальну функцію і цей добуток інтегрується в межах від -∞ до ∞:



де  - оригінал функції; *Х(ω).* - її зображення.

Перетворення Фур’є є інтегральним перетворенням, яке позначається символом: .

В системі MATLAB перетворення Фур’є оригіналу функції  здійснюється за допомогою функції:

**fourier (f)**

***Приклад 1***. Згідно інтегрального перетворення Фур’є знайти зображення F(w) функції оригіналу 

**>> syms t**

**>> f = exp(-abs(2\*t));**

**>> F=fourier (f)**

**F = 4/4+w^2**

Отже, можна записати:

.

**2. Обернене перетворення Фур’є**

В системі MATLAB обернене перетворення Фур’є здійснюється за допомогою функції:

**ifourier (F)**

де F– пряме перетворення Фур’є, t – аргумент шуканої функції оригіналу *f*(t).

***Приклад 2***. Знайти оригінал функції *f*(t), якщо її зображення.

**>> syms w**

**>> F =4./(4+w.^2);**

**>> f= ifourier (F)**

**f = exp(-2\*x)\*Heaviside(x)+exp(2\*x)\*Heaviside(-x)**

Отже, можна записати:

****

***Індивідуальні завдання***

***Завдання 1***. Згідно інтегрального перетворення Фур’є знайти зображення *Х(ω).* функції оригіналу .

|  |  |  |  |
| --- | --- | --- | --- |
| Варіант | Функція | Варіант | Зображення |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |

***Контрольні запитання та вправи***

***Лабораторна робота №9***

**ПРЯМЕ І ОБЕРНЕНЕ ВЕЙВЛЕТ - ПЕРЕТВОРЕННЯ.**

***Мета роботи:*** освоїти апарат та шляхи знаходження прямого та оберненого вейвлет- перетворення .

***Теоретичні основи***

***Контрольні запитання та вправи***

***Лабораторна робота №10***

**РОЗВ’ЯЗОК ЛІНІЙНИХ РІВНЯНЬ В СЕРЕДОВИЩІ SIMULINK**

***Мета роботи:*** здобути навіки обчислення алгебраїчних виразів та розв’язування систем лінійних алгебраїчних рівнянь згідно застосування середовища Simulink.

***Практичні вказівки та взірці виконання завдань***

**Завдання 1. Отримати сигнал в Simulink:**

.

1. Запустити Simulink. Створити вікно нової моделі. У це вікно з броузера Simulink Library Browser з вказаних папок перетягнути такі блоки:

**Sources**:

* Clock,
* Costant.

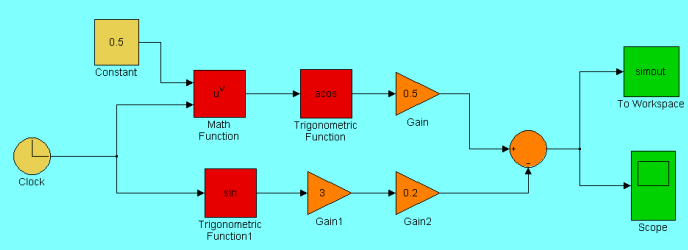
**Sinks**:

* Scope,
* To Workspace.

**Math Operations**:

* Gain,
* Math Function,
* Trigonometric Function,
* Sum.

2. Скласти наступну модель:



3. Встановити наступні параметри для блоків:

**Constant**: Constant value = 0.5.

**Math Function**: у списку параметрів Function вибрати функцію **pow** (піднесення до степеня).

**Trigonometric Function**: у списку параметрів Function вибрати функцію **acos**  - для першого блоку, **sin** – для другого блоку.

**Gain**: для параметра Gain вибрати 0.5 – для першого блоку, 3 – для другого, 0.2 – для третього.

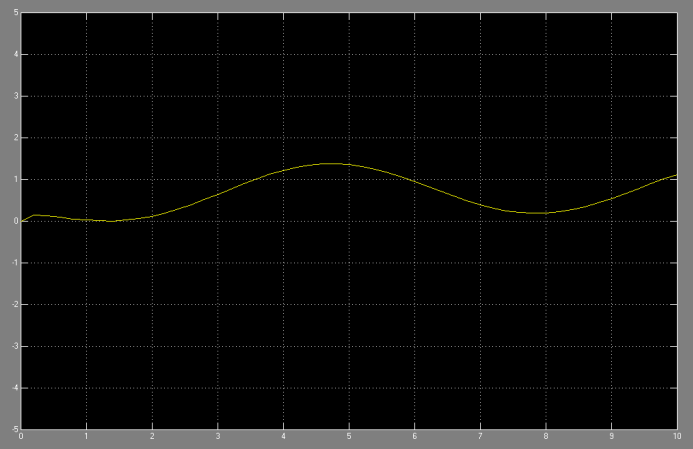
**Sum**: у вікні параметрів вибрати для List of signs: +-. Це означає, що на +буде поступати вхідний сигнал, який додаватиметься, а на – сигнал, який відніматиметься.

Готовий сигнал спрямувати на два блоки:

**Scope** – для візуального спостереження графіка сигналу;

**To Workspace** – для перенесення результатів роботи у систему MatLab.

Графік заданої функції буде виглядати наступним чином:



**Завдання 2. Ввести рівняння в програму MatLab. Побудувати графік (див. лаб.роботу 3). Порівняти отримані графіки в завданні 1 та 2.**

**Завдання 3. Розв’язати систему лінійних рівнянь засобами Simulink:**



Створити вікно нової моделі. У це вікно з броузера **Simulink Library Browser** з вказаних папок перетягнути такі блоки:

**Sources**:

* Costant – 2.

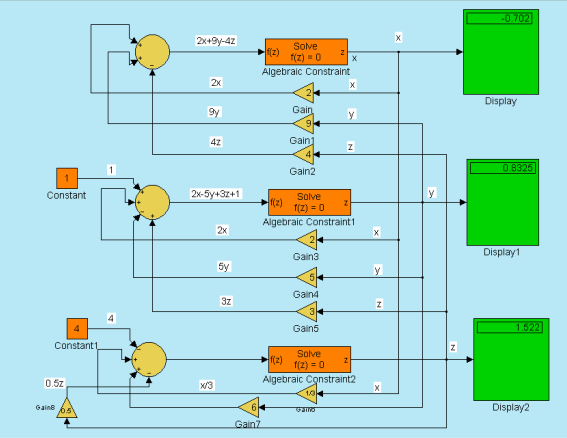
**Sinks**:

* Display – 3.

**Math Operations**:

* Algebraic Constraint – 3;
* Gain – 9;
* Sum – 3.

2. Скласти наступну модель:



Натиснувши кнопку **Start Simulation**, отримаємо розв’язки:

x = - 0,702; y = 0,8325; z = 1,522.

**ІНДИВІДУАЛЬНІ ЗАВДАННЯ**

***Завдання.*** Отримати сигнал та розв’язати систему рівнянь в Simulink:

1. . 

2. . 

3. . 

4. . 

5. . 

6. . 

7. . 

8. . 

9. . 

10. . 

11. . 

12. . 

13. . 

14. . 

15. . 

***Контрольні запитання та вправи***

***Лабораторна робота №11***

**РОЗВ’ЯЗУВАННЯ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ В СЕРЕДОВИЩАХ MATLAB ТА SIMULINK**

***Мета роботи:*** ознайомлення та використання середовищ **MATLAB та SIMULINK** для розв’язування диференціальних рівнянь

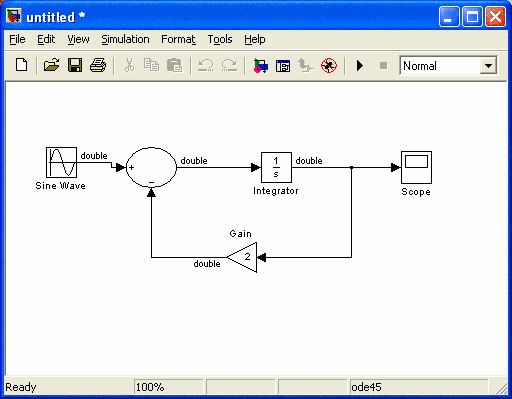
***Теоретичні і практичні вказівки***

1. **Розв’язування диференціальних рівнянь   
   у середовищі Simulink**

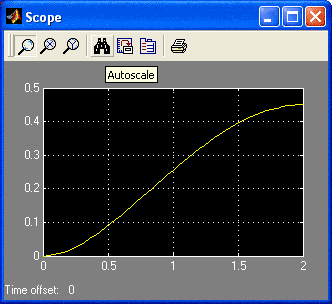
**а) Розв’язати диференціальне рівняння x'(t) + 2x(t) = sin(t) з початковими умовами x(0) = 0.**

Після запуску системи MatLab натиснути кнопку **Simulink**, а потім у вікні, що відкрилося, кнопку **Create а new Model**. У вікні, що відкрилося, створимо схему розв’язку рівняння, перетягуючи при натисненій лівій кнопці миші необхідні блоки з вікна **Simulink Library Brouser**.

Для побудови схеми розв’язку рівняння в **Simulink** використовується блок **Integrator** (підпапка **Continuos**). На його вхід подається похідна, а на виході одержують величину **x**. Блоки **Sum** (**Суматор**) і **Gain** (**Підсилювач**) (підпапка **Math**) необхідні для формування значення **x'**. Для отримання сигналу sin(t) використовується блок **Sine Wave** (підпапка **Sources**), в якому необхідно провести установки, відповідні до умов задачі, відкривши блок подвійним кліканням миші або вибравши опцію **Block Parameters** при натисненій правій кнопці миші. Отримане значення **x(t)** подається на вхід блоку **Scope**. Двічі клікнувши по даному блоку, з’являється графік розв’язку. Встановити масштаби осей, відповідні отриманому розв’язку можна, натиснувши кнопку **Autoscale**.



Мал.1



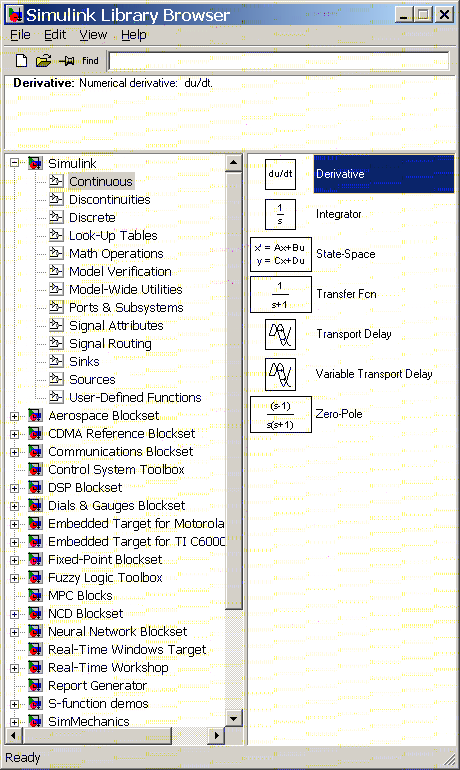
Мал.2

**Розв’язати диференціальне рівняння в програмі Simulink:**

**u′′ + 2 u′ + 5u = 0 з початковими умовами: u(0) = 1, u′(0) = 5**

Це рівняння відображає роботу випромінювача затухаючого гармонічного сигналу.

Так як змінна являється постійною (неперервною), потрібно відкрити бібліотеку **Continious** (**Постійні**), клікнувши по + зліва від іконки **Continious**:



Мал.3.

Зверніть увагу, що **u** та **u′** отримуються із **u′** та **u′′** відповідно шляхом інтегрування. Тому перетягніть з допомогою миші дві копії блоку **Integrator** (**Інтегратор**) у вікно моделі. Програма Simulink автоматично присвоїть другому блоку ім’я **Integrator 1**, щоб відрізняти його від першого блоку з іменем **Integrator**.

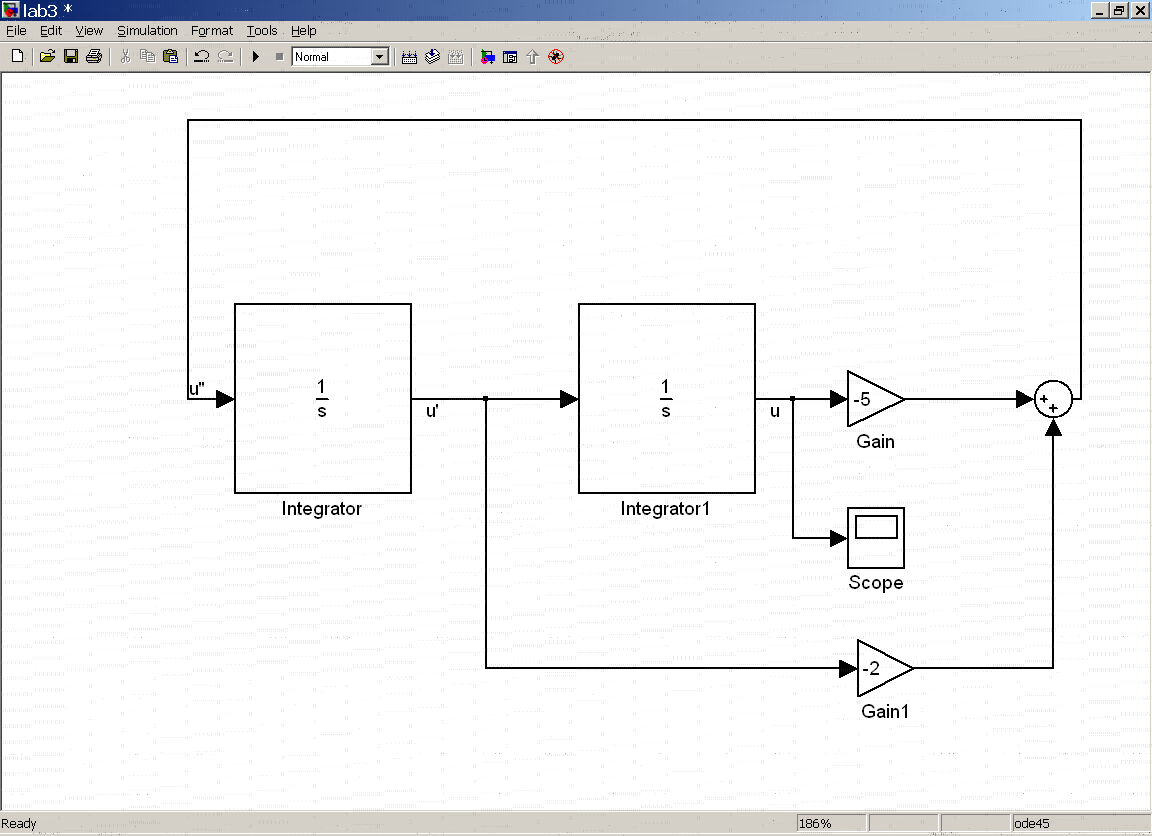
Кожний блок **Integrator** має вхідний та вихідний порти. Підрівняйте вихідний порт блоку **Integrator** з вхідним портом блоку **Integrator 1** і з’єднайте їх стрілкою, використовуючи ліву кнопку миші. Стрілка, яка з’єднує два блоки, називається **сигналом**. Двічі клікніть мишою на цій стрілці, з’явиться невелике текстове поле, в яке потрібно ввести назву **u′**.

Диференціальне рівняння можна записати у вигляді:

**u′′ = -5u – 2u′**.

Потрібно добавити інші елементи, щоб зв’язати **u′′**, ввід першого блоку ***Integrator***, з **u** та **u′** відповідно до цього рівняння.

Для цього потрібно добавити до моделі два блоки **Gain** (**Збільшення**), які виконують множення по константі, та один блок **Sum** (**Сума**), який використовується для додавання. Їх вибирають із бібліотеки **Math Operations** (**Математичні операції**).



Мал.4.

Після цього потрібно з’єднати стрілками вихід з блоку **Integrator 1** з входом блоку **Gain**, вихід з блоку **Gain** з’єднати з першим входом блоку **Sum**, а вихід з блоку **Gain 1** з другим входом блоку **Sum**.

Далі потрібно сигнал **u′**, вивід першого блоку **Integrator**, направити до порту вводу блоку **Gain 1**.

Потрібно ще до моделі добавити блок для перегляду вихідного сигналу. Відкрити бібіліотеку **Sinks** (**Приймачі**), і перетягнути копію блоку **Scope** (**Екран**) у вікно моделі. З допомогою розгалуження лінії з’єднати цей блок з лінією, яка з’єднує блоки **Integrator 1** та **Gain**.

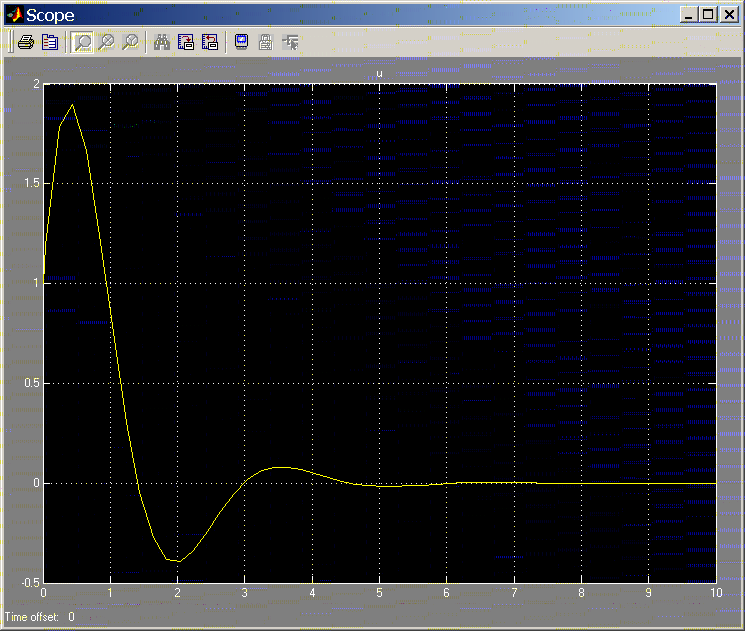
Потрібно ще відредактувати властивості блоків **Gain**. Потрібно змінити константи, на які вони перемножуються, зі значення 1 по замовчуванню на -5 для блоку **Gain** та з 1 на -2 для блоку **Gain 1**.

Необхідно задати початкові умови. Це можна зробити, двічі клікнувши курсором миші на блоках **Integrator** і змінивши рядок **Initial** **Condition** (**Початкова умова**) в діалозі **Block Parameters**.

Задамо початкові умови для **u′**: в першому блоці **Integrator** ввести значення 5, і умову для **u (0) = 1** в блок **Intgrator 1** як значення 1.

Щоб зберегти модель, потрібно використати команду меню **File/Save as**. Програма MATLAB автоматично добавляє розширення файлу **.mdl**, яке призначене для моделей.

Щоб бачити, що відбувається під час симуляції, потрібно двічі клікнути мишою на блоці **Scope**, щоб відкрився екран осцилографа, на якому буде представлений розв’язок даного диференціального рівняння.



Мал.5.

Перейти в меню **Simulation** і вибрати команду **Start**. У вікні **Scope** побачимо графік функції:

**u(t) = 3e-t sin(2t) + e-tcos(2t).**

**2. Аналітичний розв’язок диференціальних рівнянь   
з допомогою функції dsolve**

**Приклад 1.** Нехай потрібно розв’язати диференціальне рівняння другого порядку: y′′ + a2y = 0.

В командному рядку потрібно набрати:

**>> y = dsolve ('D2y + a^2\*y = 0')**

MATLAB видасть відповідь:

**Y = C1\*sin(a\*t) + C2\*cos(a\*t)**

Це означає, що загальний розв’язок даного диференціального рівняння має вигляд:

Y = C1sin(at) + C2cos(at).

Задавши початкові умови **y(0) = 0**, **y′(0) = 1**, для розв’язку даного диференціального рівняння потрібно набрати:

**>> y = dsolve ('D2y + a^2\*y = 0, y(0) = 0, Dy(0) = 1')**

Отримаємо в MATLAB відповідь:

**y = sin(at) + 2 cos(at)**

**MATH OPERATIONS – БЛОКИ МАТЕМАТИЧНИХ ОПЕРАЦІЙ**

**1. Sum – блок обчислення суми**

***Призначення***: виконує обчислення суми поточних значень сигналів.

Параметри:

* **Icon shape** – форма блоку. Вибирається із списку:
  + - round – коло;
    - rectangular – прямокутник.
* **List sign** – список знаків. В списку можна використовувати наступні знаки:
  + - + (плюс),
    - – (мінус)
    - | (роздільник знаків).
* **Saturate on integer overflow** (прапорець) – пригнічувати переповнювання цілого. При встановленому прапорці обмеження сигналів цілого типу виконується коректно.

Кількість входів і операція (складання або віднімання) визначається списком знаків параметра **List sign**, при цьому мітки входів позначаються відповідними знаками. В параметрі **List sign** можна також вказати число входів блоку. В цьому випадку всі входи будуть тими, що сумують.

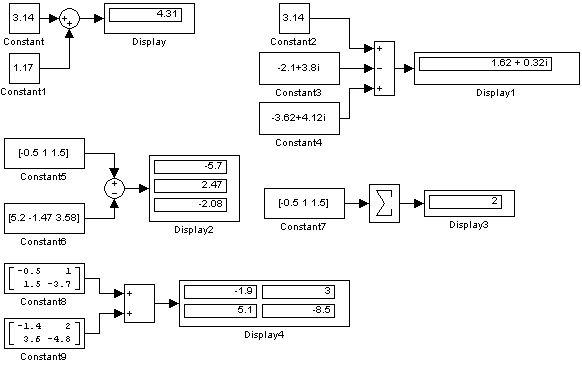
Якщо кількість входів блоку перевищує 3, то зручніше використовувати блок **Sum** прямокутної форми.

Блок може використовуватися для сумування скалярних, векторних або матричних сигналів. Типи сигналів, які сумуються, повинні співпадати. Не можна, наприклад, подати на один і той же блок сигнали цілого та дійсного типів.

Якщо кількість входів блоку більша, ніж один, то блок виконує поелементні операції над векторними та матричними сигналами. При цьому кількість елементів в матриці або векторі повинна бути одинаковою.

Якщо як список знаків вказати цифру 1 (один вхід), то блок можна використовувати для визначення суми елементів вектора.

Приклади використання блоку **Sum** показано на мал.6.



Мал.6. Приклади використання блоку **Sum**.

**2. Product** – **блок множення**

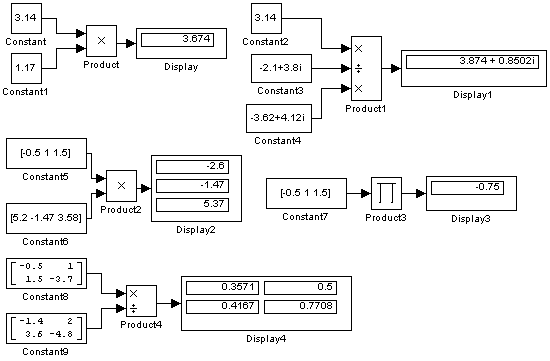
***Призначення***: виконує обчислення добутку поточних значень сигналів.

***Параметри***:

* **Number inputs** – кількість входів. Може задаватися як число або як список знаків. В списку знаків можна використовувати знаки \* (помножити) і / (поділити).
* **Multiplication** – спосіб виконання операції. Може приймати значення (із списку):
  + - **Element-wise** – поелементний.
    - **Matrix** – матричний.
* **Saturate on integer overflow** (галочка) –при встановленому прапорці обмеження сигналів цілого типу виконується коректно.

Якщо параметр **Number inputs** заданий списком, що включає окрім знаків множення також знаки ділення, то мітки входів будуть позначені символами відповідних операцій.

Блок може використовуватися для операцій множення або ділення скалярних, векторних або матричних сигналів. Типи вхідних сигналів блоку повинні співпадати. Якщо як кількість входів вказати цифру 1 (один вхід), то блок можна використовувати для визначення добутку елементів вектора.



Мал.7. Приклади використання блоку **Product** при виконанні скалярних та поелеметних операцій.

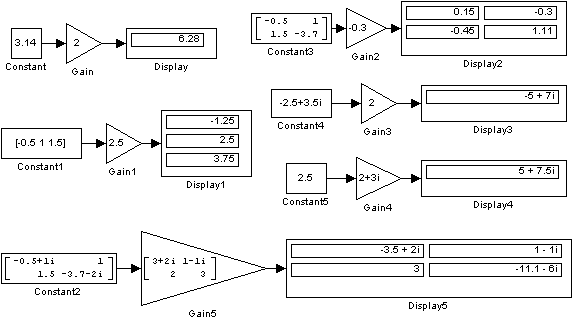
**3. Gain – підсилювач**

***Призначення***: виконує множення вхідного сигналу на постійний коефіцієнт.

***Параметри***:

* **Gain** – коефіцієнт підсилення.

Параметр блоку **Gain** може бути додатнім або від’ємним числом, як більшим, так і меншим 1. Коефіцієнт підсилення можна задавати у вигляді скаляру, матриці або вектора, а також у вигляді обчислюваного виразу.



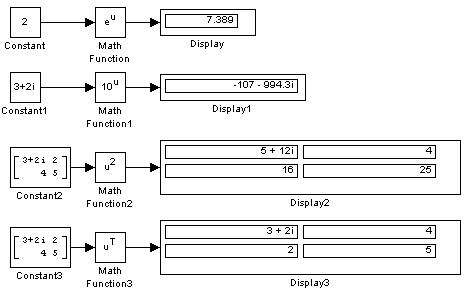
Мал.8. Приклади використання блоку **Gain**.

**4. Math Function – блок обчислення математичних функцій**

***Призначення***: виконує обчислення математичної функції.

***Параметри***:

* **Function** – вид функції, яка обчислюється (вибирається із списку):
  + - **exp** – експоненціальна функція;
    - **log** – функція натурального логарифма;
    - **10^u** – обчислення степеня 10;
    - **log10** – функції логарифма;
    - **magnitude^2** – обчислення квадрату модуля вхідного сигналу;
    - **square** – обчислення квадрату вхідного сигналу;
    - **sqrt** – квадратний корінь;
    - **роw** – піднесення до степеня;
    - **conj** – обчислення комплексно-спряженого числа;
    - **reciprocal** – обчислення приватного від розподілу вхідного сигналу на 1
    - **hypot** – обчислення кореня квадратного з суми квадратів вхідних сигналів;
    - **rem** – функція, що обчислює залишок від ділення першого вхідного сигналу на другий;
    - **mod** – функція, що обчислює залишок від ділення з урахуванням знаку;
    - **transpose** – транспонування матриці;
    - **hermitian** – обчислення ермітової матриці.
* **Output signal type** – тип вихідного сигналу (вибирається із списку):
  + - **auto** – автоматичне визначення типу;
    - **real** – дійсний сигнал;
    - **complex** – комплексний сигнал.



Мал.9. Приклади використання блоку **Math Function**.

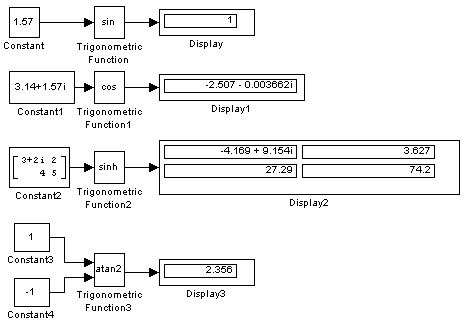
**5. Trigonometric Function – блок обчислення тригонометричних функцій**

***Призначення***: виконує обчислення тригонометричної функції.

***Параметри***:

* **Function** – вид функції, яка обчислюється (вибирається із списку): **sin**, **cos**, **tan**, **asin**, **асos**, **atan**, **atan2**, **sinh**, **cosh** і **tanh**.
* **Output signal type** – тип вихідного сигналу (вибирається із списку):
  + **auto** – автоматичне визначення типу;
  + **real** – дійсний сигнал.
  + **сomplex** – комплексний сигнал.

При векторному або матричному вхідному сигналі блок виконує поелементне обчислення заданої функції.



Мал.10. Приклади використання блоку **Trigonometric Function**.

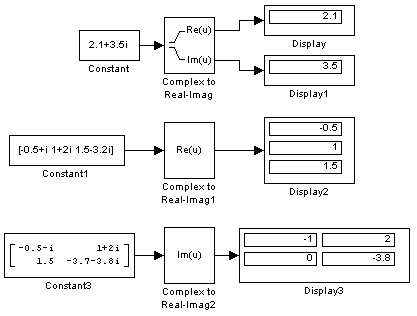
**6. Complex to Real-Imag – блок обчислення дійсної та уявної частини комплексного числа**

***Призначення***: обчислює дійсну та (або) уявну частину комплексного числа.

***Параметри***:

* **Output** – вихідний сигнал (вибирається із списку):
  + **Real** – дійсна частина;
  + **Image** – уявна частина;
  + **RealAndImage** – дійсна та уявна частина.

Вхідний сигнал блоку може бути скалярним, векторним або матричним сигналом.



Мал.11. Приклади використання блоку **Complex to Real-Imag**.

**ІНДИВІДУАЛЬНІ ЗАВДАННЯ**

***Завдання 1***. Розв’язати диференціальне рівняння 1-го порядку з початковими умовами в середовищах Simulink та MATLAB. Результати представити в графічному та в аналітичному вигляді.

***Завдання 2***. Розв’язати диференціальне рівняння 2-го порядку з початковими умовами в середовищах Simulink та MATLAB. Результати представити в графічному та в аналітичному вигляді.

***Варіант 1***

***Завдання 1***. x′ +5 x = 2 sint, x(0) = 1.

***Завдання 2***. x′′ + 4x′ + 4x = 2; x(0) = 0, x′(0) = 1

***Варіант 2***

***Завдання 1***. 5x′ +sin5(t) =3e4t, x(0) = -1.

***Завдання 2***. x′′ + 4x′+ 4x = 2sin(t); x(0) = -1, x′(0) = 1.

***Варіант 3***

***Завдання 1***. 4x′ - 2 x = 3 sint, x(0) = 2.

***Завдання 2***. 5x′′ + 3x′ + e2t = 2; x(0) = -1, x′(0) = 1.

***Варіант 4***

***Завдання 1***. -5x′ - x = 3 sin(t), x(0) = -1.

***Завдання 2***. -4x′′ + 3x′ + e5t = 3; x(0) = -1, x′(0) = 1.

***Варіант 5***

***Завдання 1***. - 4x′ +e3t = 2 sin3(t), x(0) = -1.

***Завдання 2***. x′′ + 2x′ + x = 2sint; x(0) = 0, x′(0) = 2

***Варіант 6***

***Завдання 1.*** -8x′ + 7x = x2, x(0) = 1.

***Завдання 2***. -8x′′ + 7x′ +3sin2(t) = -3; x(0) = -1, x′(0) = 1.

***Варіант 7***

***Завдання 1***. -15x′ + 6sin(t) = 5, x(0) = 1.

***Завдання 2***. -2x′′ + 3x′ - 2e3t = -4; x(0) = -1, x′(0) = 1.

***Варіант 8***

***Завдання 1***. 20x′ + 5sin(t) = 14, x(0) = -2.

***Завдання 2***. 4x′′ + 3x′ = - 2 x; x(0) = 1, x′(0) = -1.

***Варіант 9***

***Завдання 1***. -7x′ - 6sin(t)⋅x = 22, x(0) = - 4.

***Завдання 2***. -4x′′ - 5x′ = - 2 sin3(t); x(0) = -1, x′(0) = 0.

***Варіант 10***

***Завдання 1.***  5x′ + 4sin2(t) = 11; x(0) = 3.

***Завдання 2***. x′′ + 2x′ +3x = 2; x(0) = 0, x′(0) = 1.

***Варіант 11***

***Завдання 1***. 3x′ + 4sin4(t) = 8; x(0) = 1.

***Завдання 2***. x′′ - x′ = e2t sint; x(0) = -1, x′(0) = 1

***Варіант 12***

***Завдання 1***. -7x′ - 3sin3(t) = 8x; x(0) = -1.

***Завдання 2***. 8x′′ - 6x′ - e4t = 4; x(0) = 0, x′(0) = 1

***Варіант 13***

***Завдання 1***. -2x′ - 4sin4(t) = et; x(0) = 1.

***Завдання 2***. -2x′′ +5x′ - 5 = 2 sin2(t); x(0) = 0, x′(0) = 1

***Варіант 14***

***Завдання 1***. -12x′ + 4et = 2sin(t); x(0) = 3.

***Завдання 2***. 4x′′ + 2x′ = 3x; x(0) = 0, x′(0) = 1

***Контрольні запитання та вправи***

**Література**

1. **Базова**

1. Василенко І.П. та ін. Вища математика. Основні означення, приклади і задачі.Ч.2. – К.: Либідь, 1992. – 186 с.

2. Кострубій П.П. та ін. Елементи теорії функції комплексної змінної. Перетворення Фур’є і Лапласа. Збірник задач і вправ. – Львів: В-во Львівської політехніки, 2011. – 200 с.

3. Руданський Ю.К. та ін. Теорія функції комплексної змінної. Інтегральні перетворення Фур’є і Лапласа.–Львів: В-во Львівської політехніки, 2011–245 с.

4. Стадник М.М., Хом’якевич М.О. Елементи теорії функції комплекснї змінної та операційне числення. – Київ – Львів: НМЦВО, 2000. – 146 с.

1. **Допоміжна**

1. Данко П.Е., Попов А.Р., Кожевникова Т.Я. Высшая математика в упражнениях и задачах. Ч.ІІ і ІІІ. – М.: Наука, 1980. – 254 с.

2. Мартиненко В.С. Операционное исчисление. – К.: В – во Київського уныверситету, 1965. – 104 с.

3. Шелковников Ф.А., Тайкашвили К.Г. Сборник упражнений по операцыонному исчислению. – М.: Высшая школа, 1976.

1. **Методичне забезпечення**

1. Думанський О.І. Операційне числення. Конспект лекцій. – Електронний варіант, 2013. – 112 с.

2. Думанський О.І. Завдань до лабораторних робіт з дисципліни «Операційне числення». – Електронний варіант, 2011. – 72 с.

***Зміст***

**НАВЧАЛЬНЕ ВИДАННЯ**

**Чисельна реалізація задач операційного числення у середовищі Matlab. Лабораторний практикум до дисципліни „Операційне числення”**

Укладачі: **доцент Думанський Остап Іванович**

**ст.викладач Думанський Нестор Остапович**

**ст.викладач Бекас Богдан Олексійович**

**доцент Пірко Ігор Богданович**

**ст..викладач Шиманський Володимир Михайлович**