ҚАЗАҚСТАН РЕСПУБЛИКАСЫ
БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ. И. Сәтбаев атындағы
Қазақ ұлттық техникалық зерттеу университеті

Қ. Ә. Өжікенов
Р.С. Исмагулова

MATLAB МАТЕМАТИКАЛЫҚ ОРТАСЫ

Университеттің Ғылыми-әдістемелік кеңесі
оқу құралы ретінде ұсынған

Алматы 2016

ӘОЖ 519.711.3
КБЖ 22.18. я 73
Ө23

Пікір жазғандар:
Байғұншеков Ж. Ж. техн., ғыл., докторы, профессор, Қазақстан Республикасы Ұлттық Ғылым академиясының академигі;
Үтепбергенов Е. Т. техн., ғыл., докторы, профессор;
Қосболов С. Б. техн., ғыл., докторы, профессор.

Қазақстан Республикасы Білім және ғылым министрлігінің 2016 жылғы жоспары бойынша басылды.

Өжікенов Қ.Ә.

Ө23 MATLAB математикалық ортасы. Оқу құралы\ Өжікенов Қ.Ә., Исмагулова Р.С. – Алматы: Қ. И. Сәтбаев атындағы ҚазҰТЗУ, 2016. – 150 стр. Сурет -28. Кесте – 15. Библиогр., тізімі - 29 атау.
ISBN 978-601-228-946-6

Бұл оқу құралында MATLAB бағдарламалық қолданбалы пакетінен алғашқы мағлұматтарды бере отырып, осы ортада сызықтық алгебра есептерін, сызықтық және сызықтық емес дифференциалдық теңдеулерді шешу үшін есептеулер жүргізудің әдістерінен мағлұматтар берілген.

Оқу құралы «Аспап жасау», «Автоматтандыру және басқару» мамандықтары бойынша білім алып жатқан студенттерге арналған. Авторлар ұсынылып отырған бұл оқу құралының MATLAB бағдарламалық қолданбалы пакетін өз бетінше игергісі келетін оқырмандарға да пайдасы тиер деген сенімде.

ISBN 978-601-228-946-6 © Өжікенов Қ.Ә., Исмагулова Р.С., 2016
© Қ. И. Сәтбаев атындағы ҚазҰТЗУ, 2016

КІРІСПЕ

Компьютерлік технологияның қарқынды дамуына байланысты техникалық жүйелерді жобалаудағы күрделі мәселелерді шешу жолдарына да үлкен өзгерістер енуде.

MATLAB, DesignLab, WorkBench, OrCad және т.б. қолданбалы пакеттері әр түрлі техникалық жүйелерді зерттеудегі және жобалаудағы инженерлік қолданыстарды кеңейтіп, сапалы өзгерістерге қол жеткізуге мүмкіндік береді. Қолданыс аумағының шексіз болуынан қолданбалы пакеттердің ішіндегі бірден бір болашағы зор болып MATLAB пакеті саналатыны сөзсіз. Сондықтан да соңғы жылдары жоғары және кәсіптік орта оқу орындарының оқу жоспарларында жүйелерді модельдеудің мәселесіне көбірек уақыт бөлетін болды. Компьютерлік технологияның соңғы жетістіктерін қолдана отырып, студенттердің техникалық пәндерді оқуы кезінде де, инженердің техникалық жүйелерді жобалауы кезінде де жүйелердегі өтетін үдерістер физикасын және осы үдерістерге зерттеулер жүргізу үшін қолданылатын математикалық тәсілдерді тереңдетіп білулері талап етіледі. Үдерістер физикасын жеткілікті деңгейде білу – нақты жүйеге сай модель құруға және модельдеу нәтижелерінің нақты болуына кепіл бола алады.

MATLAB пакеті жайында төл тілімізде жарық көріп отырған бұл оқу құралы заман талабына сай жазылып, жаратылыстану мен техникалық салаларда мемлекеттік тілімізде білім алу барысында жүйелерді модельдеуге бағытталған пәндерді оқытуда қолдану үшін шығарылып отыр. Қазақ тілінде жазылған бұл оқу құралы MATLAB пакеті жайындағы бүкіл мағлұматтарды бірден толық қамтып береді деу, әрине, артықтау болар, дегенмен де авторлар осы пакетті қолданудың әдіс-тәсілдерін үйренудегі ең басты да, негізгі деген мағлұматтарды қамтуға тырысты.

Оқу құралында теориялық мағлұматтарды берумен қатар, оларға көптеген нақты мысалдарды қолдану қарастырылған.

Сонымен қатар, мемлекеттік тілімізге әлі де техникалық терминдердің толық аударылып және мемлекеттік стандартпен бекітіліп енбегендіктен, кітаптың соңында қолданылған терминдердің аудармалары келтірілген.

Авторлар қазақ тілінде жарық көріп отырған бұл оқу құралын, төл тілімізде «Аспап жасау», «Автоматтандыру және басқару» мамандықтары бойынша дәріс алып жатқан студенттеріміз үшін, MATLAB ортасында бағдарламалар құру және осы ортада жүйелерді модельдеудің қыры мен сырын меңгеру жолында көмекші құрал бола алады деген сенімде.

1. MATLAB ОРТАСЫ. КОМАНДАЛЫҚ РЕЖИМДЕГІ ЖҰМЫС

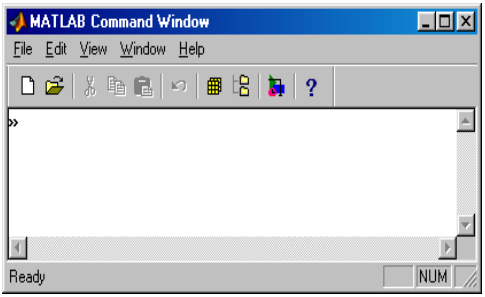
1.1. MATLAB ортасы

Жұмыс істеу кезінде инженердің де, ғалымның да, тіпті қарапайым студенттің де математикалық есептеулерге жүгінуіне тура келетін кездері болады. Ал мұндай есептеулер бір жағдайларда қарапайымдау болса, енді бір жағдайларда күрделі де болып келеді. Ал күрделі есептерді шығаруда MATLAB (MATrix LABoratory) сияқты математикалық пакеттерді қолдануға болады.

MATLAB программалау тілі ретінде Нью-Мексико университетінің компьютер ғылымдары факультетінің деканы болған Клив Моулермен (ағылш. Cleve Moler) 1970- жылдардың соңында жасалды. Оны жасаудағы мақсат – факультет студенттеріне ФОРТРАН-ды оқытпай-ақ, Linpack және EISPACK бағдарламалық кітапханаларын қолданудың мүмкіндігін жасау болды. Көп ұзамай-ақ бұл жаңа тіл басқа университеттерге де тарап кетеді де, қолданбалы математика саласындағы ғалымдардың да аса қызығушылығына тап болады. 1983 жылы инженер Джон Литтл (ағылш. John N. (Jack) Little) Клив Моулермен Стэнфорд университетіндегі кездесуінде осы тілмен таныс болып, бұл тілдің болашағының зор боларын білген ол Клив Моулермен және Стив Бангертпен (ағылш. Steve Bangert) бірігеді. Біріккен күшпен олар MATLAB-ты C тілінде қайта жазып, оны ары қарай дамыту мақсатында 1984 жылы The MathWorks компаниясын құрады. Бастапқы уақытта MATLAB тек басқару жүйелерін жобалауда ғана қолданылады да, артынан басқа да ғылыми және инженерлік қолданыстарда болады.

MATLAB негізінде матрицалық есептеулерді жүргізу үшін арналған сұхбаттық орта болып жасалған болатын (MATrix LABoratory). Уақыт өте келе оның графикалық жүйесі дамытылып, арнайы кластағы есептеулерді жүргізу үшін командалар кітапханаларымен (Toolboxes және т.б.) және Maple-дің компьютерлік алгебра ортасымен толықтырылады. Қазіргі таңда MATLAB ғылыми зерттеулер жүргізудегі ең мықты құралдың бірі болып отыр.

MATLAB-тағы барлық жұмыс бағдарламаны matlab.exe арқылы қосқанда пайда болатын командалық терезе (Command Window) арқылы іске асырылады. 1.1 - суретте MATLAB жүйесінің командалық терезесі көрсетілген.



1.1-сурет. MATLAB жүйесінің командалық терезесі

Онда мәзір жолы, құрал-саймандар жақтауы, жұмыс аймағы мен күй жолы бар. Жұмыс аймағында командалар енгізіледі және есептеу нәтижелері шығарылады. Командаларды енгізу «>>» түріндегі командалық жолға шыққан шақыру белгісі арқылы іске асырылады. Терезенің төменгі жағында орналасқан күй жолына жүйенің деректері шығарылып тұрады.

1.2. Командалық режимдегі жұмыс

MATLAB-та есептеуді бағдарламалық режимде немесе командалық режимде жұмыс істеуге болады. Бағдарламалық режимде MATLAB тілінде жазылған арнайы бағдарлама шақырылады. Бұл бағдарлама арқылы алғашқы деректер енгізіліп, есептеулер жүргізіліп, соңынан олардың нәтижелерін шығару іске асырылады. Командалық режимде командалар мен есептелінетін өрнектер командалық терезедегі енгізу жолдары арқылы беріледі. Есептеу нәтижелері командалық терезеге шығарылады. Мысалы, *4+5* өрнегін командалық терезеге енгізіп, пернетақтадағы Enter-ді басқанда компьютер экранында келесідей нәтижені көруге болады:

```
» 4+5
ans =
    9
»
```

Өрнектің есептелген мәні ans (answer) арнайы айнымалысына меншіктеледі де, жеке жолға шығарылады. *Ans* айнымалысында басқа айнымалыға меншіктелмеген соңғы есептелген өрнектің мәні сақталады. Ал, бұл мәнді әрі қарай жүргізілетін есептеулерге де қолдануға болады:

```
» 4+5
ans =
    9
» ans/3
ans =
    3
»
```

Егер командалық жолға енгізілген өрнектің соңына «;» (нүктелі үтір) белгісін қойса, онда есептеу нәтижесі шығарылмайды. Мысалы (бағанның оң жағында сол командалар «;» белгісінсіз енгізілген):

```
» 4+5;
» ans/3;
» ans
ans =
    3
»
```

```
» 4+5
ans =
    9
» ans/3
ans =
    3
» ans
ans =
    3
»
```

Сонымен қатар «;» белгісін командалық жолда берілетін өрнектерді ажырату үшін де қолдануға болады:

```
» g=8; h=5;k=g>h;
» g
g =
    8
» h
h =
    5
» k
k =
    1
»
```

Командалық жолға енгізілген командалар мен өрнектердің бәрі де сақталады. Оларды қайтара енгізу командаларды тік және кері бағытта көруге мүмкіндік беретін ↑ және ↓ батырмалары арқылы іске асырылады.

Командалық терезені тазарту үшін *clc* командасы қолданылады.

Егер енгізілетін өрнек бір жолдың бойына симайтын болса, онда оларды келесі жолға көшіру үшін «…» көпнүктелік белгісі қойылады (үш немесе одан да көп):

```
» 3+...
5-7+9/...
3
ans =
    4
»
```

MATLAB-та жұмыс сеансы сессия деп аталады. Сессия енгізу-шығару жолдарынан және ескертулер мен қателіктер жайындағы деректерді шығаратын жолдардан тұрады. Қателіктер жайындағы деректер алдын ала үш сұрақ белгісімен беріледі де, есептеу тоқтатылады:

```
» 1h=5
??? 1
   |
Missing operator, comma, or semi-colon.
»
```

Ескертулер алдын ала Warning сөзімен беріледі. Ескерту есептеуді тоқтатпайды, ол тек қана қолданушыға табылған қатенің кейінгі есептеу нәтижелеріне кері әсер етуінің мүмкіндігін ғана ескертеді:

```
» 1/0
Warning: Divide by zero.
ans =
  Inf
»
```

1.3. Математикалық өрнектер

Математикалық өрнектерге сандық тұрақтылар, айнымалы атаулары, функциялар, арифметикалық операторлар белгілері, дөңгелек жақшалар жатады.

1.3.1. Нақты сандар

Сандарды жазуда олардың бүтін бөлігін бөлшек бөлігінен ажырату үшін «.» (нүкте) белгісі қолданылады. Мантиссаны ретінен ажырату үшін «*e*» белгісі қолданылады. Мысалы:

```
7    -2    5.8    2.456e-5    9.914e12
```

1.3.2. Комплекстік сандар

MATLAB-та нақты сандармен қатар комплекстік сандармен де есептеулер жүргізуге болады. ($\sqrt{-1}$) жорамал бірлігін белгілеу үшін *i* немесе *j* атты айнымалылар қолданылады. Алгебралық түрдегі комплекстік сандарды енгізгенде жорамал бөлігіндегі сан тұрақтысы мен жорамал бірлігінің арасына көбейтінді белгісін қоймаса да болады:

```
» 7+5i
ans =
  7.0000 + 5.0000i
```

Ал егер жорамал бөлік айнымалы болса, онда комплекстік сандарды енгізу кезінде оның жорамал бөлігі мен жорамал бірлігінің арасына көбейту белгісін қою керек:

```
» 7
ans =
    7
» 5+ans*i
ans =
  5.0000 + 7.0000i
»
```

Комплекстік сандармен жұмыс істеу функциялары 1.3.8 тарауда берілген.

1.3.3. Айнымалылар

Айнымалылар сандық, белгілік, матрицалық немесе векторлық түрде болуы мүмкін. Айнымалы атауларының келесідей қалыптасу тәртіптері бар:

1. Айнымалы атауы кез келген белгі санынан тұрады, бірақ оның алғашқы 31-і ғана сақталады және анықталады.

2. Атау басқа айнымалылар мен функциялардың атауларымен ұқсас болмауы керек.

3. Атау әріптен, саннан және астыңғы сызық белгісінен тұруы мүмін.

4. Атау әріптен басталу керек.

5. Айнымалы атауларында бас әріп пен кіші әріптің айырмашылығы бар.

Айнымалы атауын қолданылуының дұрыстығын тексеру үшін isvarname айнымалы атауы түрде шақырылатын арнайы функция қолданылады. Бұл функция егер айнымалы атауы дұрыс болса *1* мәнін қабылдайды, ал кері жағдайда *0* мәнін қабылдайды. Мысалы:

```
» isvarname alfa
ans =
    1
» isvarname 1a
ans =
    0
»
```

Айнымалыға мәнді меншіктеу үшін меншіктеу операциясы қолданылады: айнымалы атауы = өрнек. Мысалы:

```
» a=5
a =
    5
»
```

Айнымалы типі айнымалыға берілген өрнек арқылы анықталады. Айнымалының мәнін экранға шығару үшін оның атын командалық жолға жазып, Enter-ді басу керек.

1.3.4. Жүйелік айнымалылар

MATLAB-та келесідей жүйелік айнымалылар қолданылады:

i не j - жорамал бірлік,
realmax - модулі бойынша ең жоғарғы нақты сан (1.7977e+308),
realmin - модулі бойынша ең төменгі нақты сан (2.2251e-308),
inf - шексіздік (нөлге бөлудің нәтижесі),
ans - басқадай айнымалыға берілмеген соңғы есептелген өрнектің мәні,
NaN - сан емес. Анықталмаған нәтижені көрсетеді (0/0 немесе Inf/Inf амалдарының нәтижесінде шығады),
pi - π саны,
eps - жылжымалы нүктесі бар санмен жүргізілген амалдың дәлсіздігі (1.0 саны мен келесі жақын тұрған жылжымалы нүктесі бар санның аралығы 2^{-52}-ге тең).

Жүйелік айнымалылар жүйемен беріледі, бірақ қайтара анықталуы мүмкін.

1.3.5. Арифметикалық амалдар

Нақты және комплекстік сандарға қосу, алу, көбейту, бөлу және дәрежелік амалдар қолдануға болады (олар, тиісінше, «+», «-», «*», «/», «^» белгілерімен көрсетіледі).

Басымдық тәртібі бойынша арифметикалық операциялардың орындалу реті келесідей түрде болады:

1. Дәрежелеу (^).
2. Көбейту мен бөлу (*, /).
3. Қосу мен алу (+, -).

Басымдықтары бірдей амалдар солдан оңға қарай орындала береді. Амалдардың орындалу реттілігін дөңгелек жақшаларды қолдану арқылы да өзгертуге болады.

Өрнектің мәнін есептеу үшін оны алдымен командалық жолға енгізіп, пернетақтадан Enter-ді басу керек:

```
» 4^2-5/(2+3^2)
ans =
  15.5455
»
```

Комплекстік сандарды есептеуде операндтар дөңгелек жақшаға алынады:

```
» 4+3i-(5-7i)/(3+6i)
ans =
  4.6000 + 4.1333i
»
```

Комплекстік-түйіндес сандарды есептеуде дәйекше (‘) қолданылады:

```
» d=7-4i
d =
  7.0000 - 4.0000i
» d'
```

```
ans =
  7.0000 + 4.0000i
»
```

1.3.6. Қатынас амалдары

Операндтарды салыстыру үшін келесідей қатынас амалдары қолданылады:

==	- тең,
~=	- тең емес,
<	- кіші,
>	- үлкен,
<=	- кіші немесе тең,
>=	- үлкен немесе тең.

Қатынас амалдарының нәтижесі «ақиқат» немесе «жалған» болады. MATLAB-та «ақиқат» *1* саны арқылы, ал «жалған» *0*саны арқылы белгіленеді.

Қатынас амалдарының басымдығы арифметикалық амалдарға қарағанда төмен болады.

1.3.7. Логикалық амалдар

Логикалық амалдар және оларға сәйкес функциялары 1.1-кестеде келтірілген.

1.1-кесте

Логикалық амалдар және функциялар

Амал	Атауы	Функция
&	логикалық ЖӘНЕ	and
|	логикалық НЕМЕСЕ	or
~	логикалық ЕМЕС	not
	НЕМЕСЕ-ні шығару	xor

Логикалық амалдар орындалғанда нөлге тең емес операндтар «ақиқат» болып, ал олардың нөлге тең болғандары «жал-

ған» болып саналады. Логикалық операторлар мен функциялардың жұмысы 1.2 кестеде келтірілген.

1.2-кесте

Логикалық операторлар мен функцияларының жұмысы

Операндтар		Логикалық операторлар мен функциялар			
X	y	x&y and(x,y)	x|y or(x,y)	~x not(x)	xor(x,y)
1	1	1	1	0	0
1	0	0	1	0	1
0	1	0	1	1	1
0	0	0	0	1	0

Логикалық амалдардың басымдықтары қатынас амалдарына қарағанда аса төмен саналады. Ал ЖӘНЕ логикалық амалының басымдығы НЕМЕСЕ амалына қарағанда жоғары саналады.

1.3.8. Элементар функциялар

MATLAB функциялары кірістірілген және сыртқы болып жіктеледі. Кірістірілген элементар функциялар тізімі 1.3-кестеде келтірілген. Сыртқы функциялар .m кеңейтілуімен жеке файлдарда сақталады. Сыртқы функциялар 4 тарауда қарастырылады.

Функцияны шақыру үшін оның атынан кейінгі дөңгелек жақшада жазып көрсету керек:

```
» sin(pi/2)
ans =
    1
»
```

1.3-кесте

MATLAB-тың кірістірілген элементар функциялары

Функция атауы	Қызметі	Функция атауы	Қызметі
Тригонометриялық			
sin(x)	синус	*asin(x)*	арксинус
sinh(x)	гиперболалық синус	*asinh(x)*	гиперболалық арксинус
cos(x)	косинус	*acos(x)*	арккосинус
cosh(x)	гиперболалық косинус	*acosh(x)*	гиперболалық арккосинус
tan(x)	тангенс	*atan(x)*	арктангенс
tanh(x)	гиперболалық тангенс	*atanh(x)*	гиперболалық арктангенс
sec(x)	Секанс	*asec(x)*	арксеканс
sech(x)	гиперболалық секанс	*asech(x)*	гиперболалық арксеканс
csc(x)	Косеканс	*acsc(x)*	арккосеканс
csch(x)	гиперболалық косеканс	*acsch(x)*	гиперболалық арккосеканс
cot(x)	Котангенс	*acot(x)*	арккотангенс
coth(x)	гиперболалық котангенс	*acoth(x)*	гиперболалық арккотангенс
atan2(y,x)	$(-\pi, \pi)$ аралығындағы *y/x*-тің арктангенсі		
Экспоненциалдық, көрсеткіштік және логарифмдік			
exp(x)	экспонента	*log(x)*	натуралдық логарифм
log10(x)	ондық логарифм	*nextpow2(x)*	*x*-ке тең немесе үлкен болатын жақын санды алу үшін 2 санын дәрежелеу
sqrt(x)	квадрат түбірі		
pow2(x)	2 санының *x* дәрежесі	*log2(x)*	негізі 2 болатын логарифм
Комплекстік сандармен жұмыс істеу функциялары			
real(x)	комплекстік санның нақты бөлігі	*abs(x)*	комплекстік санның модулі
imag(x)	комплекстік санның жорамал бөлігі	*angle(x)*	комплекстік санның аргументі
complex(a, b)	*a+jb* комплекстік санын құрайды	*conj(x)*	комплекстік түйіндес санды *x*-ке кері қайтарады
Дөңгелектеу және бөлуден қалдықты ажырату			
fix(x)	нөлге қарай бағыттағы жақын тұрған бүтінге қарай дөңгелектеу	*floor(x)*	теріс шексіздіктегі жақын тұрған бүтінге қарай дөңгелектеу
ceil(x)	оң шексіздіктегі жақын тұрған бүтінге қарай дөңгелектеу	*round(x)*	жақын тұрған бүтінге қарай дөңгелектеу
mod(x,y)	таңбаны есепке алуымен бүтін санды бөлуден қалдық	*rem(x,y)*	модуль бойынша бүтін санды бөлудегі қалдық
sign(x)	санның таңбасын анықтау		

Тригонометриялық функциялардың аргументі радианмен берілу керек, ал кері тригонометиялық функциялар да нәтижелерін радианмен шығарып береді.

Элементар функциялар нақты және комплекстік аргументтермен жұмыс істеуге арналған.

Help elfun командасы қысқа деректерімен элементар функциялардың барлық тізімін шығарып береді. Функция туралы толық дерек алу үшін help функция атауы командасы қолданылады. Анықтама жүйесін қолдану 1.6 тарауда қарастырылады.

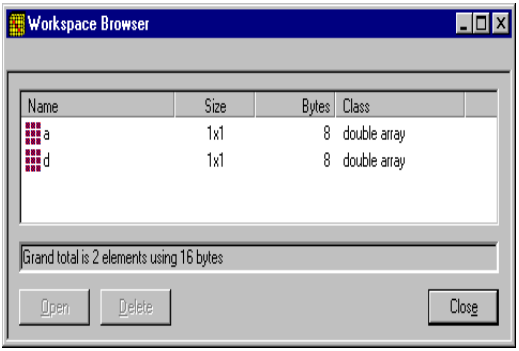
1.4. Жұмыс аймағы

Жұмыс істеу кезінде құрылған айнымалылар жадының жұмыс аймағына (workspace) орналасады. Жұмыс аймағындағыларды көру үшін who және whos командалары қолданылады. Who командасы қолданыстағы айнымалылардың аттарын экранға шығару үшін, ал whos командасы экранға жиымдардың өлшемдері және айнымалылардың жадыдағы шамасы мен типтері жайындағы деректерді экранға шығару үшін қолданылады:

```
» a=5
a =
     5
» d=2
d =
     2
» who
Your variables are:
a        d
» whos
  Name      Size            Bytes  Class
  a         1x1                 8  double array
  d         1x1                 8  double array
Grand total is 2 elements using 16 bytes
»
```

Сонымен қатар, енгізу жолына workspace командасын теріп шақыру арқылы, немесе  (Workspace Browser) батырмасын

басу арқылы, не болмаса File/Show Workspace командасын таңдау арқылы Workspace Browser терезесінде (1.2-сурет) жұмыс аймағын көруге болады.



1.2-сурет. Workspace Browser терезесі

Workspace Browser терезесінде айнымалыны бөліп алып, сонан соң Delete батырмасын басу арқылы өшіруге де болады, ал егер Open батырмасын таңдаса, онда Editor/Debugger (редактор/дұрыстаушы) терезесінде бөліп алған айнымалының мәнін көруге болады.

Жұмыс аймағын толық тазарту үшін clear командасы қолданылады. Бірнеше таңдамалы айнымалыларды жою үшін clear айнымалы атауларының тізімі («босорын» белгісі арқылы жазылады) командасы қолданылады. Мысалы:

```
» who
Your variables are:
j      q      r      u      y
» clear q
» who
Your variables are:
j      r      u      y
» clear u j
» who
Your variables are:
r      y
```

```
» clear
» who
»
```

Айнымалылардың енгізілуі мен жойылуы шараларынан жұмыс аймағы бөлшектенген күйге түседі. Ал бұл, өз кезегінде, есептеу жылдамдығының төмендеуіне алып келеді. Жұмыс аймағын жинақтау pack командасын орындау арқылы іске асырылады. Жұмыс аймағын сақтау save командасы арқылы орындалады:

save файл атауы	жұмыс аймағындағы атауы көрсетілген және .mat кеңейтілуімен берілген барлық айнымалыларды файлға жазу
save файл атауы айнымалы атауларының тізімі («босорын» белгісі арқылы жазылады)	жұмыс аймағындағы барлық айнымалыларды файлға жазу

Ертеректе сақталған жұмыс аймағын жүктеу load командасы арқылы іске асырылады: load файл атауы немесе load файл атауы айнымалы атауларының тізімі («босорын» белгісі арқылы жазылады).

Айнымалыларды әр түрлі файлдардан таңдап алып жүктеуге болады. Load командасының екінші бір нұсқасы айнымалыларды таңдамалы түрде жүктеуде қолданылады.

Жұмыс аймағын сақтау және оны жүктеу үшін, тиісінше, File/Save Workspace As және File/Load Workspace командалары қолданылуы мүмкін. Жұмыс аймағын жазу және жүктеу командаларын қолдану мысалдары төменде келтірілген:

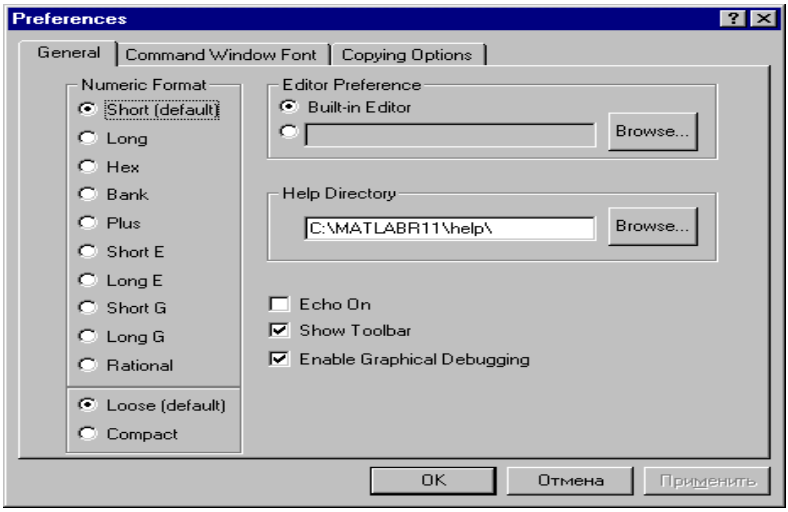
```
» save fg
» what
MAT-files in the current directory C:\MATLABR11\work
fg
» clear
```

```
» who
» load fg
» who
Your variables are:
ans      h      u
»
```

Келтірілген мысалдағы what командасы ағымдағы каталогындағы файлдардың атауларын шығару үшін қолданылды.

1.5. Жүйе параметрлерін күйге келтіру

File/Preferences командасы параметрлерді күйге келтіретін сұхбат терезесін шақыруға мүмкіндік береді. Терезеде үш қыстырма бар (1.3-сурет).



1.3-сурет. Preferences терезесінің General қыстырмасы

General қыстырмасының өрістерінің қызметтері 1.4-кестеде келтірілген.

1.4-кесте

General қыстырмасының өрістерінің қызметтері

Өріс	Қызметі
Numeric Format	сандарды көрсетудің қалыбын таңдау (1.5– кесте) және нәтижелер қорытындысының түрлері: loose – еркін (нәтиже қорытындысы бос жолдармен кезектеседі) және compact (шағын, бос жолсыз шығару).
Editor Preference	қолданылатын мәтіндік редакторды беру (үнсіздік бойынша салынып қойылған)
Help Directory	анықтама файлдары орналасатын каталог
Echo On	Script-файл сценарийдің орындау командаларын экранға шығару керек-керек еместігін анықтау
Show Toolbar	құрал-саймандар жақтауын көрсету
Enable Graphical Debugging	Графиктің дұрыстауын қолдану

1.5-кесте

Сандарды көрсетудің қалыптары

Қалыбы	Қалыптың сипаттамасы	Мысал
short	бекітілген нүктесі (5 таңбалы) бар қалыптағы қысқаша сандарды көрсету	1.2345
long	бекітілген нүктесі (15 таңбалы) бар қалыптағы көп сандарды көрсету	0.00045634487671
hex	он алтылық есептеу жүйесіндегі сандарды көрсету	3f3de8325237f974
bank	ондық нүктеден кейін екі таңбамен көрсету (ақшалар үшін қолданылады)	1.23
plus	санның тек таңбасы ғана шығарылады (+ немесе -)	+

1.5-кестенің жалғасы

short e	жылжымалы нүктесі бар қалыптағы сандарды қысқаша көрсету (мантиссаның 5 таңбасы және реттің 3 таңбасы)	1.2345e+000 4.5634e-004
long e	жылжитын нүктесі бар қалыптағы көп сандарды толық көрсету (мантиссаның 15 таңбасы және реттің 3 таңбасы)	4.56344876713453e-004
short g	MATLAB бекітілген немесе жылжитын нүктемен берілген сандарды көрсетуде қысқаша қалыпты таңдайды	1.2345
long g	MATLAB бекітілген немесе жылжитын нүктемен берілген сандарды көрсетуде ұзын қалыпты таңдайды	1.234567890123456
rational	рационалды баған түрінде көрсету	10/3

Сонымен бірге, сандарды көрсету қалыбын format қалып атауы командасымен де беруге болады. Мысалы, format long. Экран бетіне барлық түсінікті қалыптағы деректерді шығару үшін help format командасын пайдалану қажет.

Command Window Font қыстырмасында командалық терезенің әріптерінің параметрлері беріліп, ал Copying Options қыстырмасында көшірме опциялары таңдап алынуы мүмкін.

1.6. Анықтама жүйесін қолдану

MATLAB бойынша мәліметтерді келесідей әдістер арқылы алуға болады:

– help командасы;
– lookfor командасы;
– HELP мәзірі;
– PDF қалыбы;
– The Math Works фирмасының Web-желісі.

Функцияның қолданылуы туралы анықтамалық мәліметті help функция атауы командасын енгізу арқылы алуға болады.

Дерек мәтінінде функция атаулары кіші әріптермен жазылғанмен, нақты функцияда бас әріптермен жазылу керек екенін естен шығармау қажет.

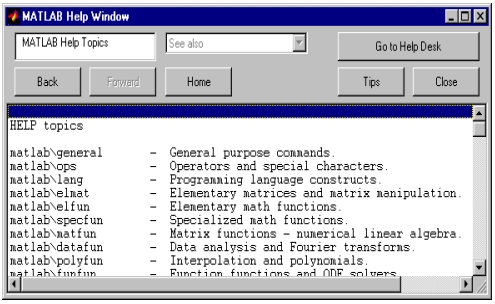
MATLAB функциялары қолданылуы бойынша топтастырылған. Help командасы экран бетіне функция топтарының тізімін экранға шығарады. Нақты топ бойынша мәлімет алу үшін help топтың атауы командасы қолданылады. lookfor түйінді сөз немесе lookfor 'түйінді сөз тіркесі' командасы түсініктеменің бірінші қатарындағы түйінді сөз бойынша М-функциясын іздеу үшін қолданылады. Түйінді сөз бойынша табылған жол экран бетіне шығарылады. All – бұл опцияның қосылуы. Ал lookfor түйінді сөз – all немесе lookfor 'түйінді сөз тіркесі' – all түсініктеменің толық мәтінінің шығуын қамтамасыз етеді. Help мәзірінің командаларының қызметі 1.6–кестеде келтірілген.

1.6-кесте

Help мәзірінің командаларының қызметі

Команда	Қызметі
Help Window	дерек терезесі
Help Tips	анықтама алуға арналған мәлімет терезесі
Help Desk (HTML)	қатты дискіге орналастырылған анықтама жүйесіне рұқсат алу
Examples and Demos	мүмкіндіктерді көрсету
About MATLAB	орнатылған нұсқа туралы мәлімет

Help Window терезесін (1.4-сурет) Help/Help Window, Help/Help Tips командалары арқылы немесе құрал-саймандар жақтауындағы  батырмасымен, не болмаса helpwin командасы арқылы шақыруға болады. Қажетті бөлімге көшу үшін тышқанмен оның тақырып атауына екі рет басу қажет немесе helpwin бөлім аты командасын қолдану керек. Back, Forward, Home батырмалары белсенді алдыңғы, келесі және бастапқы анықтамалық терезелеріне көшуге мүмкіндік береді.



1.4-сурет. Help Window терезесі

Tips батырмасы еске салу терезесін экранға шығарады. Бұл режимде lookfor, which, demo және general командалары арқылы See also тізімі белсенді болады.

Help/Help Desk (HTML) командасы HTML қалыбындағы анықтама дерегіне қол жеткізуді қамтамасыз етеді. Сол терезеде PDF қалыбындағы ақпаратқа да қол жеткізуге арналған батырма бар.

Бақылау жұмысы.

MATLAB-ты калькулятор ретінде қолдану. MATLAB-тағы қарапайым есептеулер жүргізу

Жұмыстың мақсаты:

1. MATLAB пакетінің интефейсімен және негізгі жағдайларымен танысу.

2. Мәліметтердің қарапайым түрлерімен, оларды енгізумен және есептеу нәтижелерін көрсету жолдарымен танысу.

3. Диалогтық режимде жұмыс істеудің тәжірибелік жолдарын игеру.

Жұмыстың тапсырмасы.

1 есеп. MATLAB интерфейсімен танысу.

2 есеп. MATLAB демонстрациялық мысалдарымен танысу.

3 есеп. Калькулятор режимінде келесі іс-әрекеттерді істеу керек:

- Бастапқы операндтарды енгізу.
- 1 және 2 операндтармен 1-ші операцияны істеу.
- Нәтижемен және 1 операндпен 2 операцияны істеу.
- Нәтижемен және 2 операндпен 3 операцияны істеу.
- 1 операндтың әр мүшесімен 3-тік дәрежесін орындау.

Тапсырмалар варианты

№	1операнд	2 операнд	Операторлар		
			1	2	3
1	V=[12 34 61 45 11]	v = 34	*	./	+
2	V=[80 67 34 11 45]	v = 43	/	.*	-
3	V=[19 77 45 11 67]	v = -5	+	.\	/
4	V=[11 98 67 45 22]	v = 7	-	.*	/
5	V=[67 34 67 45 56]	v = -12	+	.\	*
6	V=[18 36 45 45 4]	v = 10	/	./	-
7	V=[55 43 8 45 23]	v = 44	/	.*	/
8	V=[32 28 55 45 34]	v = 87	*	-	/
9	V=[14 34 33 45 15]	v = 78	*	+	+
10	V=[15 23 17 45 9]	v = -22	/	-	*
11	V=[10 34 10 45 7]	v = -14	*	-	*
12	V=[95 56 5 45 54]	v = 99	+	./	+
13	V=[18 90 35 45 46]	v = 32	*	.*	-
14	V=[24 34 87 45 88]	v = -43	/	.*	/
15	V=[14 41 90 45 77]	v = 55	/	+	+

Бақылау сұрақтары

1. MATLAB редакторлық терезесінің құрылымы қандай?
2. Айнымалы аттарын жасау тәртібі.
3. Команданы енгізу тәртібі.
4. Функциялар мен операндтарды енгізу тәртібі.
5. Өрнектерді енгізу тәртібі.
6. Операциялардың қорытындыларын көру тәртібі.
7. Сандар қандай форматта беріледі?
8. MATLAB-та анықтама жүйесін қалай қолдануға болады?

2. ЖИЫМДАРМЕН ЖҰМЫС

MATLAB-та барлық деректер жиым түрінде қолданылады. Негізінен, әрбір айнымалы жиым түріндегі шама ретінде алынады, мысалы, *R=45* жазуы бір элементтен тұратын сан, мәні 45-ке тең *R*-векторы екендігін білдіреді. Мысалы, 1.2 – суреттен Workspace Browser терезесінде берілген *a* және *d* айнымалыларының 1x1 өлшемімен көрсетілгендігін байқауға болады.

MATLAB-та бір өлшемді (вектор-баған немесе вектор-жол), екі өлшемді (матрица) және көпөлшемді жиымдармен де жұмыс істеуге болады. Жиымда элементтерді нөмірлеу 1-ден басталады. Жиым өлшемі – бұл әр өлшем бойындағы элементтер саны.

2.1. Векторлар мен матрицалардың қалыптасуы

Вектор-жолды енгізу [вектордың атауы]=[үтір не «босорын» арқылы берілген вектор элементтерінің мәні] түрінде жүзеге асырылады.

Вектор-бағанның элементтерін енгізу үшін нүктелі үтірмен бөлу немесе әрбір элементтің мағынасының енгізілуінен кейін Enter батырмасын басу қажет, яғни

```
» a=[2 7 3 9 4];
» s=[3;5;7];
» h=[3
4]
h =
    3
    4
» a
a =
    2    7    3    9    4
» s
s =
    3
    5
    7
»
```

Матрица элементтерінің жолын енгізу кезінде «босорын» немесе үтір арқылы бөлуге болады, ал жолдары бір-бірінен нүктелі үтір немесе Enter батырмасын басу арқылы бөлінеді.

```
» s=[1,2,3; 4 5 6
7 8,9]
s =
    1    2    3
    4    5    6
    7    8    9
»
```

Комплекстік сандар мен формулалар жиым элементтері бола алады.

```
» h=[8*sin(0.54) 3-9 sqrt(-1) 3+4i]
h =
  4.1131        -6.0000      0 + 1.0000i   3.0000 + 4.0000i
»
```

A, B, C, D түрінде берілген бірнеше жиымдар немесе векторлардың көлденеңінен (бірігетін жиымдардың жол саны өзара тең болуы керек) бірігуі келесі түрде жүзеге асырылады: [*A,B,C,D*] немесе [*A B C D*].

A, B, C, D жиымдарының тігінен (бірігетін жиымдардың баған саны өзара тең болуы керек) біріктіру үшін нүктелі үтірмен немесе Enter батырмасын басу арқылы оларды бөлу қажет.

Біріккен жиымдардың мысалы:

```
» a=[1 2;3 4];
» b=[5 6 7;8 9 10];
» d=[11 12 13];
» c=[8 5 6 4];
» f=[a b]
f =
    1    2    5    6    7
    3    4    8    9   10
» h=[d c]
h =
```

```
   11    12    13     8     5     6     4
» k=[d;b
d]
k =
   11    12    13
    5     6     7
    8     9    10
   11    12    13
»
```

Тізім мәнінен басқа жиым элементтері келесі түрде берілген: бастапқы мән : адым : соңғы мән. Егер адым шамасы бірге тең болса, онда оны бермеуге де болады. Мысалы, элементтері бірлік адыммен *3*-тен *7* аралығында жататын *k* векторы қалыптассын. *P* матрицасының әрбір жолы *3*-тік адыммен және әртүрлі бастапқы және ақырғы мәндермен берілген. *U* матрицасын бергенде аралық бірінші жолдың алғашқы төрт элементін және екінші жолдың соңғы бес элементін береді.

```
» k=[3:1:7]
k =
    3     4     5     6     7
» P=[0:3:9; 1:3:10]
P =
    0     3     6     9
    1     4     7    10
» U=[0:3:9 100 30; 200 1:2:9 ]
U =
    0     3     6     9   100    30
  200     1     3     5     7     9
»
```

Егер вектордың барлық элементтері аралықпен берілсе, квадрат жақшаларды қолданбауға болады:

```
» w=8:4:32
w =
   8   12   16   20   24   28   32
»
```

Бірлік адымды көрсетпеуге болады:

```
» q=3:10
q =
   3   4   5   6   7   8   9   10
»
```

2.2. Жиым элементіне сілтеме

Вектор элементтеріне сілтемеде бір индекс қолданылады (вектор-жолға және вектор-бағанға да бірдей) және ол вектордың атауынан кейін дөңгелек жақшада жазылады, мысалы, *g (7)* - *g* векторының жетінші элементі.

Матрица элементтеріне сілтемеде дөңгелек жақшада үтір арқылы жазылатын екі индекс қолданылады – баған нөмірі және жол нөмірі, мысалы, k (3,7) - үшінші жолдың және жетінші бағананың қиылысуында орналасқан k матрицасының элементі.

Есептелген жиым элементеріне қарапайым мәндерді енгізуге болады. Осы кезде талап ететін өлшемдегі матрица автоматты түрде құрылады. Мәнге иеленбеген басқа элементтер нөлге тең болады:

```
» r(3,2)=7
r =
   0   0
   0   0
   0   7
» r (5,1)=5
r =
   0   0
   0   0
   0   7
   0   0
   5   0
»
```

Жоғарғы мысалда өлшемі *3x2* болатын матрица автоматты түрде құрылды, ал кезекті мәндерді берумен оның өлшемі үлкейтілді де, ол *5x2* болды.

Жиым элементтерінің мәндерін беруде жиымның төменгі оң жақ бұрышында орналасқан элементтен бастаған дұрысырақ болады. Бұлай элементтерді қосу жиымның өлшемін өзгертпейді (жиымға арналған жад бөлу бір рет орындалады).

Матрица элементтеріне матрицаның бірінен кейін бірі тұрған баған элементтері бойынша алынған бірөлшемді жиымдағы орындарын анықтайтын бір индексті қолдану арқылы сілтеме жасауға болады. Мысалы, *f(2,2)* және *f(6)* өрнектері *f* жиымының бір элементіне ғана жасалған сілтеме екенін көрсетеді:

```
» f=[1 2 ;3 4; 5 6; 7 8]
f =
    1    2
    3    4
    5    6
    7    8
» f(2,2)
ans =
    4
» f(6)
ans =
    4
»
```

2.3. «Қос нүкте» операторын қолдану

Жиым элементтерінің индекс аралықтарын беру үшін «:» («қос нүкте») операторы қолданылады. Бұл оператордың көмегімен «қос нүкте» жиымның бөлігін ерекшелеуге болады, мысалы, *m(in:ik,jn:jk*) өрнегі m матрицасынан *in*-нен *ik*-ға дейінгі жолды, *jn*-нен *jk*-ға дейінгі бағанды бөліп алады. Мысалы:

```
» h=[1 2 3 ;4 5 6;7 8 9]
h =
   1   2   3
   4   5   6
   7   8   9
» h(2:3,1:2)
ans =
   4   5
   7   8
»
```

М матрицасында бағаны мен жолының көрсетілуі қос нүкте операторы арқылы келесідей жүргізіледі: М(жол нөмірі,:) және М(:, баған нөмірі). Мысалы:

```
» d=[1 2 3; 4 5 6; 7 8 9]
d =
   1   2   3
   4   5   6
   7   8   9
» d(:,2)
ans =
   2
   5
   8
» d(1,:)
ans =
   1   2   3
»
```

Қос нүкте операторын соңғы жол нөмірі орнына (бағанның) қолдануда end сөзін қолдануға болады. Төменгі мысалда экранға жиымның екінші жолынан бастап екінші бағанның соңына дейінгі матрица үзіндісі шығарылады.

```
» d=[1 2 3; 4 5 6; 7 8 9]
d =
   1   2   3
   4   5   6
   7   8   9
» d(2:end, 2:end)
ans =
   5   6
   8   9
»
```

2.4. Берілген жиымдағы жолды және бағанды өшіру

Берілген жиымда жолды және бағанды өшіруде бос жиым қолданылады (қос квадрат жақша []). Бос жиым өлшемі 0x0. Берілген жиымдағы жолды және бағанды өшіруде бос жиым енгізу қажет.

```
» d=[1 2 3; 4 5 6; 7 8 9]
d =
   1   2   3
   4   5   6
   7   8   9
» d(1,:)=[]
d =
   4   5   6
   7   8   9
»
```

Бірден бірнеше жолды немесе бағанды өшіруге болады. Төменгі мысалда бірінші және екінші бағандар өшірілген.

```
» d=[1 2 3; 4 5 6; 7 8 9]
d =
   1   2   3
   4   5   6
   7   8   9
```

```
» d(:,1:2)=[]
d =
    3
    6
    9
»
```

2.5. Векторлар мен матрицаларға элементар функцияларды қолдану

MATLAB-та элементар функцияларға жиымдар жатады. Мұндай жағдайда функция аргумент өлшеміне сай өлшемдегі жиымды кері қайтарады. Қайтарылатын жиымның әр элементі жиымның элементі – аргументіне сәйкес функция мәніне тең болады. Мысалы:

```
» h=[1 2 3 ;4 5 6;7 8 9]
h =
    1    2    3
    4    5    6
    7    8    9
» sqrt(h)
ans =
   1.0000   1.4142   1.7321
   2.0000   2.2361   2.4495
   2.6458   2.8284   3.0000
»
```

2.6. Матрицалық және векторлық амалдар

Матрица және векторлық амалдар (2.1-кесте) векторлық және матрицалық есептеу ережелерімен сәйкес.

2.1-кесте

Матрицалық және векторлық амалдар

Амал	Сипаты	Мысал
+	бірдей өлшемдегі матрицалардың (векторлардың) қосындыларын есептеу. Амал нәтижесі жиым өлшеміне сәйкес, әрбір элемент келесі жиым элементтеріне лайықты	» b=[1 2 3;4 5 6]; » c=[7 2 4; 4 8 3]; » b+c ans = 8 4 7 8 13 9 »
-	бір өлшемдегі жиымдардың әр түрлілігі	» c=[7 2 4; 4 8 3]; » b=[1 2 3;4 5 6]; » b-c ans = -6 0 -1 0 -3 3 »
*	матрицаларды көбейту, векторлардың скалярлық көбейтіндісі (егер сол операнд бағандарының саны оң операнд жолдың санымен бірдей болса, онда жиымдарды көбейтуге болады)	» a=[1 2;3 4]; » b=[1 2 3;4 5 6]; » a*b ans = 9 12 15 19 26 33
‘	(дәйекше) - нақты элементтері бар жиымдардың тасымалдануы, жиымдардың комплекстік элементтермен комплекстік түйіндесуі және тасымалдануы	» c=[7 2 4; 4 8 3] c = 7 2 4 4 8 3 » c' ans = 7 4 2 8 4 3 » » d d = 3.0000 + 1.0000i 3.0000 - 7.0000i » d' ans = 3.0000 - 1.0000i 3.0000 + 7.0000i

2.1-кестенің жалғасы

.’	жиымдарды тасымалдау	» d=[3.0000 + 1.0000i 3.0000 - 7.0000i] d = 3.0000 + 1.0000i 3.0000 - 7.0000i » d.' ans = 3.0000 + 1.0000i 3.0000 - 7.0000i » c=[5 6 7];c.' ans = 5 6 7 »
^	матрицаны дәрежелеу (матрицалардың өз-өзіне көбейтуі жүргізіледі). матрицаны (-1)-ге дәрежелеу кері матрицаны береді. Сонымен қатар, Inv (A) функциясы кері (A) матрицасын есептеу үшін арналған. матрицаны (-n)-ге кері дәрежелеу, ол кері матрицаны өз-өзіне n рет көбейтуіне тең.	» a=[2 3; 6 7] a = 2 3 6 7 » a^3 ans = 206 255 510 631 » a*a*a ans = 206 255 510 631 » a^-1 ans = -1.7500 0.7500 1.5000 -0.5000 » ans*a ans = 1 0 0 1 » a^-2 ans = 4.1875 -1.6875

		-3.3750 1.3750 » inv(a) ans = -1.7500 0.7500 1.5000 -0.5000 »

Егер қосу, азайту, көбейту немесе бөлу амалдарының бірі немесе бөлу амалының операнды сан болса, жиымның әрбір элементіне және санға лайықты амал орындалады. Мысалы, *A* матрицасының және *2* санының қосындысын тауып, алынған матрицаны *5*-ке бөлу керек:

```
» A=[1 2; 3 4]
A =
   1   2
   3   4
» A+2
ans =
   3   4
   5   6
» ans/5
ans =
  0.6000   0.8000
  1.0000   1.2000
»
```

d және *q* векторларының векторлық көбейтіндісін есептеу үшін (үш элементтен тұрады) *cross(d,q)* функциясы қолданылады:

```
» d=[1 2 3];
» q=[4 5 6];
» cross(d,q)
ans =
  -3   6   -3
»
```

Бірөлшемді *d* және *q* векторларының скалярлық көбейтіндісін табу *dot(d,q)* функциясы арқылы іске асырылады. Оның

қолданылуы кезінде векторлардың элементтерінің саны мен бағандар саны және жолдар санының комбинациясы бірдей болулары керек:

```
» a=[1 2 3 4];
» b=[4 3 2 1];
» dot(a,b)
ans =
   20
» a*b
??? Error using ==> *
Inner matrix dimensions must agree.
» a*b'
ans =
   20
»
```

Бұл мысалдағы *a*b* амалы қатеге алып келді. Себебі, екі операнд та жолдық. Ал енді *b* векторын тасымалдаудан кейін қателіктің шығуына ешқандай да себеп болмайды.

2.7. Жиымдарға элементтері бойынша орындалатын амалдар

Матрицалар мен векторларға элементтер бойынша амал (2.2–кесте) тек бір өлшемді жиымдарда орындалады (егер олардың екі операнды да жиым болса). Нәтижесінде бір өлшемдегі және бір типтегі жиым болады.

2.2-кесте

Матрица және вектормен элементтер бойынша амалдар

Амал	Сипаты	Мысал
.*	Бір жиымды келесі жиымға элементтері бойынша көбейту. Әрбір элементке қатысты жиым элементінің көбейтіндісі қорытынды жиым болып табылады	» a=[1 4;2 5] a = 1 4 2 5 » d=[1 2 ; 3 4] d = 1 2 3 4 » a.*d ans = 1 8 6 20 »
./	Сол жақтағы операндты оң жақтағы операндқа элементтер бойынша бөлу	» a./d ans = 1.0000 2.0000 0.6667 1.2500
.\	Оң жақтағы операндты сол жақ операндқа элементтер бойынша бөлу	» a.\d ans = 1.0000 0.5000 1.5000 0.8000
.^	Элементтер бойынша дәрежелеу	» a.^d ans = 1 16 8 625 » a.^2 ans = 1 16 4 25 » a^2 ans = 9 24 12 33 »

Ешқандай цикл қолданбай, жиымдардың элементтері бойынша амалы, бір ғана жалғыз формуламен жиым элементтерінің мәндерін есептеуге мүмкіндік береді. Мысалы, функция мәндерінің кестесін есептеу үшін қос нүкте операторы арқылы вектор аргументтерін құруға болады. Содан кейін, матрицалық амалдар мен элементар функцияларды қолдана отырып, элементтер бойынша векторлық функцияларды есептеуге болады. Мысалы,

$$f(x) = \frac{\cos(2x)}{\sqrt{1 + x \cdot \sin^2(x)}}$$

функциясының мәндерінің кестесін *5*-тен *7*-ге дейінгі аралықта *0.5* адыммен есептеу керек:

```
» x=5:0.5:7
x =
   5.0000    5.5000    6.0000    6.5000    7.0000
» f=cos(2*x)./sqrt(1+x.*sin(x).^2)
f =
  -0.3546    0.0023    0.6964    0.7956    0.0682
»
```

2.8. Жиыммен жұмыс істеу функциялары

2.3–кестеде жиымдарды түрлендіру мен функцияларды құру көрсетілген. Егер функцияны шақыру кезінде жиымның шамасы көрсетілсе, шақыру функциясы бір аргументпен квадратты матрицаны құрады.

2.3–кесте

Жиыммен жұмыс істеу функциялары

Функция	Орындалатын әрекет	Мысал
zeros(n,m) *zeros(n)*	Нөлдік элементтермен *n*x*m* өлшемдік матрица құру. Бұны орындау үшін, оң жақтағы төменгі бұрышты *0*-ге теңестіреміз. Егер функцияны шақыру кезінде бір аргумент берілсе, онда квадраттық матрица қалыптасады	*» s=zeros(2,2)* *s =* *0 0* *0 0* *» h(3,2)=0* *h =* *0 0* *0 0* *0 0* *»*
ones(n,m) *ones(n)*	Әрбір элементі *1*-ге тең *n*x*m* өлшемді матрица құру	*» ones(2,3)* *ans =* *1 1 1* *1 1 1* *»
eye(n,m) *eye(n)*	Бас диагональдарының элементі *1*-ге, қалғаны *0*-ге тең *n*x*m* өлшемді матрица құру	*» eye(3)* *ans =* *1 0 0* *0 1 0* *0 0 1* *» eye(2,3)* *ans =* *1 0 0* *0 1 0*
rand(n,m) *rand(n)*	*0* мен *1*-дің арасында бірдей орналасқан кездейсоқ сандардың *n*x*m* өлшемді матрицасын құру. Егер бір ғана аргумент берілсе, *n*-реттік квадраттық матрица құрылады	*» rand(2,3)* *ans =* *0.8913 0.4565* *0.8214* *0.7621 0.0185* *0.4447* *»
randn(n,m) *randn(n)*	Нормалдық заңдылықпен орналасқан (математикалық күтумен - *0* және квадрат ауытқумен - *1*) элементтері кездейсоқ сандар болатын матрица құру. Аргументтері *rand* функциясына сәйкес беріледі	*» randn(2)* *ans =* *-0.4326 0.1253* *-1.6656 0.2877* *»
magic(n)	Әрбір баған, жол және диагональ элементтерінің қосындылары өзара тең болатын *n*-реттік квадрат матрицасын – сиқырлы матрица құру	*» magic(3)* *ans =* *8 1 6* *3 5 7* *4 9 2*

2.3-кестенің жалғасы

diag(A) *diag(A,k)*	Егер *A* функциясының аргументі ретінде матрица болса, функция бас диагональ элементтерінің векторын қайтарады. Егер *A* функциясының аргументі ретінде вектор болса, функция негізгі диагональ элементтері нақты вектор элементтеріне тең болатын диагональдық матрицаны кері қайтарады. Егер функцияны шақыруы кезінде екі параметр берілсе, екінші параметр диагональ нөмірін анықтайды (санау нүктесі – негізгі диагональ, ол *0* тең. Диагональдан жоғары орналасқан элементтер оң нөмірге ие, диагональдан төмені - теріс)	*» f=[1 2 3; 4 5 6; 7 8 9]* *f =* *1 2 3* *4 5 6* *7 8 9* *» diag(f)* *ans =* *1* *5* *9* *» g=[11 22 33];* *» diag(g,-2)* *ans =* *0 0 0 0 0* *0 0 0 0 0* *11 0 0 0 0* *0 22 0 0 0* *0 0 33 0 0* *»*
cat(q, A, B, C)	*A,B,C* матрицаларының аралық вертикаль *(q=1)* немесе горизонталь *(q=2)* бойынша қосылуы	*» f=[1 2 3];* *» k=[4 4 5];* *» cat(1,f,k)* *ans =* *1 2 3* *4 4 5* *» cat(2,f,k)* *ans =* *1 2 3 4 4 5* *»*
fliplr(A)	*A* матрицаларының бағандарын кері ауыстырады (матрица бағандарын салыстырмалы вертикаль ось бойынша ауыстыру)	*» d=[1 2 3; 4 5 6; 7 8 9]* *d =* *1 2 3* *4 5 6* *7 8 9* *» fliplr(d)* *ans =* *3 2 1* *6 5 4* *9 8 7*
flipud(A)	*A* матрицасының жолдарын кері алмастыру	*» d=[1 2 3; 4 5 6; 7 8 9]* *d =* *1 2 3* *4 5 6*

2.3-кестенің жалғасы

		7 8 9 » *flipud(d)* *ans* = 7 8 9 4 5 6 1 2 3 »
rot90(A)	*A* матрицасын сағат тіліне қарсы 90°-қа бұру	» *d* *d* = 1 2 3 4 5 6 7 8 9 » *rot90(d)* *ans* = 3 6 9 2 5 8 1 4 7 »
inv(A)	Кері матрицаны есептеу	» *a=[2 3; 6 7]* *a* = 2 3 6 7 » *inv(a)* *ans* = -1.7500 0.7500 1.5000 -0.5000 »
det(A)	Квадраттық матрицаның анықтауышын есептеу	» *a=[2 3; 6 7]* *a* = 2 3 6 7 » *det(a)* *ans* = -4 »
ndims(A)	Көпөлшемді жиымның сандық өлшемін есептеу	» *s=[4 5; 7 8]* *s* = 4 5 7 8 » *ndims(s)* *ans* = 2
size(A)	Бірінші элементі *A* матрицасының жолдарының саны, екіншісі - бағандарының саны болатын	» *h=[2 3 4; 4 5 6]* *h* = 2 3 4

2.3-кестенің жалғасы

	векторды қайтарады	*4 5 6* *» s=size(h)* *s =* *2 3* *» s(1)* *ans =* *2* *» s(2)* *ans =* *3* *»*
length(V)	*V* вектор элементтерінің санын қайтарады	*» s=[1 8 3 7]* *s =* *1 8 3 7* *» length(s)* *ans =* *4* *»*
sum(A) *sum(A,k)*	*A* векторы элементтерінің қосындысын қайтарады. Егер *A* матрица болса, функция әрбір баған элементтерінің қосындыларының векторын қайтарады. Функция екі аргументпен шақырылуы мүмкін: бірінші - жиым, екінші аргумент жол (*k=2*) немесе баған элементтерінің қосындыларын есептейді	*» s=[1 4 3 7];sum(s)* *ans =* *15* *» h=[1 2 3; 4 5 6];sum(h)* *ans =* *5 7 9* *» h=[1 2 3; 4 5 6];sum(h,2)* *ans =* *6* *15* *»*
prod(A) *prod(A,k)*	*A* векторы элементтерінің көбейтіндісін қайтарады. Егер *A* матрица болса, функция әрбір баған элементтерінің көбейтіндісінің векторын қайтарады. Екінші параметр *sum* функциясы секілді, жиым жолының және бағаның көбейтіндісін есептейді.	*» s=[1 4 3 7];prod(s)* *ans =* *84* *» h=[1 2 3; 4 5 6];prod(h)* *ans =* *4 10 18* *»*
sort(A) *sort(A, k)*	Вектор элементтерін (матрица бағандарының элементтерін (*k=1* немесе екінші аргумент жоқ болса) немесе матрица жолдарының элементтерін (*k=2*) өсуі бойынша	*» f=[3 6 1 3];sort(f)* *ans =* *1 3 3 6* *» h=[7 3 5; 1 5 2]*

2.3-кестенің жалғасы

max(A, [], k)	Матрица жолдарының ең аз (ең көп) элементтерін (*k=2*) немесе оның бағандарының (*k=1*) элементтерін іздейтін үш параметрі бар функциясының шақыру түрі Функция қайтарылатын екі параметрмен шақырылуы мүмкін: біріншісі - ең аз (ең көп) элемент немесе олардың векторы, екіншісі - ең аз (ең көп) элементтер Егер ең аз (ең көп) вектордағы элементтер (жолға, бағанға) бірнеше функция бірінші элемент көрсеткіштерін қайтарады	*ans =* *1* *» [a,b]=min(f)* *a =* *1* *b =* *3* *» h* *h =* *7 3 5* *1 5 2* *» [a,b]=min(h)* *a =* *1 3 2* *b =* *2 1 2* *» max(h,[],2)* *ans =* *7* *5* *»
any(A) *any(A, k)*	егер берілген вектора бір нөлдік емес элемент болса, онда 1-ді қайтарады. Егер аргумент матрица болса, функция векторды қайтарады, элементтері матрица бағандарымен сәйкес. Функция екі параметрмен шақырылуы мүмкін: біріншісі - жиым, екіншісі - бағандарды анықтайды (*k=1*) немесе жолдарды (*k=2*) анықтайды, ол үшін нөлдік емес элементтердің бар-жоғын анықтау керек	*» c=[0 0; 4 6; 2 0]* *c =* *0 0* *4 6* *2 0* *» any(c)* *ans =* *1 1* *» any(c,2)* *ans =* *0* *1* *1* *»
all(A) *all(A, k)*	егер берілген вектора нөлдік элемент жоқ болса, онда 1-ді қайтарады Егер аргументі матрица болса, функция элементтері матрица бағандарымен сәйкес векторды қайтарады. Функция екі параметрмен шақырылуы мүмкін: біріншісі - жиым, екіншісі - бағандарды анықтайды (*k=1*) немесе жолдарды (*k=2*) анықтайды, ол үшін нөлдік емес	*» c* *c =* *0 0* *4 6* *2 0* *» all(c)* *ans =* *0 0* *» all(c,2)* *ans =* *0*

2.3-кестенің жалғасы

	элементтердің бар-жоғын анықтау қажет	*1* *0* »
reshape(A,n,m)	А жиымының баған элементтерінен тізбектелу жолымен құрылған *n*x*m* өлшемді жиымын қайтарады	*» f=[2 4; 5 6; 4 5]* *f =* *2 4* *5 6* *4 5* *» s=reshape(f,1,6)* *s =* *2 5 4 4 6 5*
repmat(A,n,m)	А блоктың *n* вертикаль және *m* горизонталь бойынша екі өлшемді жиымды қайтарады	*a =* *1 2* *3 4* *» A=repmat(a,2,2)* *A =* *1 2 1 2* *3 4 3 4* *1 2 1 2* *3 4 3 4*
k=find(x) *k=find(x,шарт)* *[i,j]=find(X)* *[i,j]=find(X,шарт)*	*k=find(x)* және *[i,j]=find(X)* нөлдік емес *x* бірөлшемді және екі өлшемді *X* жиым элементерінің индекстерін қайтарады *k=find(x,шарт)* және *[i,j]=find(X,шарт)* шартты қанағаттандыратын *x* бірөлшемді және екі өлшемді *X* жиым элементтерінің индекстерін қайтарады	*» a* *a =* *1 2 -7* *8 4 -23* *56 0 3* *» [i,j]=find(a>5)* *i =* *2* *3* *j =* *1* *1* »

Функцияларды өңдеудің толық тізімі help datafun командасымен шақырылуы мүмкін.

2.9. Сызықтық теңдеулер жүйелерін шешу

$A \cdot x = B$ (мұндағы A - коэффициенттер матрицасы, B - бос мүшелердің векторы, x - шешімдер векторы) түріндегі сызықтық теңдеулер жүйелерін шешуде $x = A \setminus B$ амалын қолданады.

Сонымен қатар, сызықтық теңдеулер жүйесін коэффициенттер матрицасын айналдыру жолымен шешуге де болады:

$x = inv(A) * B$. Айта кетер бір жайт - ол шешімнің рационалдық болмайтындығы.

Мысалы, теңдеулердің жүйесін шешейік.

$$\begin{cases} 2 \cdot x_1 + 4 \cdot x_2 + 7 \cdot x_3 = 3 \\ 4 \cdot x_1 + 5 \cdot x_2 + 8 \cdot x_3 = 5 \\ 3 \cdot x_1 + 5 \cdot x_2 + x_3 = 1 \end{cases}$$

```
» A=[2 4 7; 4 5 8; 3 5 1]
A =
    2    4    7
    4    5    8
    3    5    1
» B=[3; 5; 1]
B =
    3
    5
    1
» x1=A\B
x1 =
    1.0444
   -0.5111
    0.4222
» x2=inv(A)*B
x2 =
    1.0444
   -0.5111
    0.4222
» A*x1
ans =
    3.0000
    5.0000
    1.0000
»
```

B/A амалы B*inv(A) өрнегіне эквивалентті болады.

2.10. Жиымдарға қатынас және логикалық амалдарды қолданудың ерекшеліктері

Қатынас амалының операнды ретінде сандар да, жиымдар да бола алады. Салыстырылатын жиымдар бір өлшемді болулары қажет. Салыстыру амалының нәтижесінде бірлер мен нөлдерден тұратын өлшемі операндыларға сәйкес жиым алынады (салыстыру нәтижесіндегі әрбір элемент салыстырылатын жиым элементіне сәйкес болады):

```
» a=[1 5; 2 6]
a =
   1   5
   2   6
» b=[4 6; 1 4]
b =
   4   6
   1   4
» a>b
ans =
   0   0
   1   1
»
```

Егер операндтардың біреуі скаляр, ал екіншісі жиым болса, онда скаляр әрбір жиым элементімен салыстырылады. Алынған нәтиже шамасы жиым-операндтың мөлшерімен сәйкес болатын жиым болады:

```
» a
a =
   1   5
   2   6
» a<=2
ans =
   1   0
   1   0
»
```

Комплекстік элементтері бар жиымдарға >, >=, <, <= амалдарын орындаған кезде, жиымдардың комплекстік элементтерінің тек нақты бөліктері ғана салыстырылады. Ал == және ~= амалдарын орындаған кезде, нақты және жорамал бөліктері ескеріледі:

```
» c=[3+1i 3-7i]
c =
  3.0000 + 1.0000i   3.0000 - 7.0000i
» d=[1-3i 3-7i]
d =
  1.0000 - 3.0000i   3.0000 - 7.0000i
» c>d
ans =
    1    0
» c==d
ans =
    0    1
»
```

Бірөлшемді матрицаларда логикалық амалдар матрицаның әр элементіне орындалады (1.2–кесте):

```
» a=[1 3 0 0];b=[0 1 3 2]; a&b
ans =
    0    1    0    0
» a|b
ans =
    1    1    1    1
»
```

2.11. Амалдарды орындаудағы басымдықтар

Төменде амалдарды орындаудағы басымдықтардың реті көрсетілген:

1. Дөңгелек жақша.

2. Тасымалдау (.’), комплекстік түрде тасымалдау (‘), дәрежелеу (^), элементтік дәрежелеу (.^).

3. Қосылғыш (+), азайтқыш (-). Логикалық терістеу (~).
4. Көбейту және бөлу (.* , ./ , .\ , * , / , \ ).
5. Қосу (+) және азайту (-).
6. Амалдар арасындағы байланыс (>, >=, < , <=, == , ~=).
7. Логикалық ЖӘНЕ (&).
8. Логикалық НЕМЕСЕ (|).

2.12. Матрицалармен жұмыс істеудің мысалдары

1-мысал. Матрицаның қосымша диагоналының элементтерін кемуі бойынша орналастыру керек.

```
»A=[1 2 7; 8 4 23; 56 3 9]
A =
    1    2    7
    8    4   23
   56    3    9
»A1=rot90(A);D=diag(A1);D1=flipud(sort(D));B1=diag(D);
C1=diag(D1);A1=A1-B1+C1;
»A=rot90(rot90(rot90(A1)))
A =
    1    2   56
    8    7   23
    4    3    9
»
```

2-мысал. Бірөлшемді *A* жиымы берілген. Барлық оң элементтерді *P* жиымына, ал теріс және нөлдік элементтерді *N* жиымына орнату керек. *P* және *N* жиымдарын қалыптастыру кезінде элементтерінің алғашқы орналасу ретін сақтау керек. Алынған жиымдардың элементтер санын экранға шығару керек.

```
» A=[1 2 -7 8 4 -23 56 0 3 9]
A =
    1    2   -7    8    4  -23   56    0    3    9
»
at=A.';at(:,2)=at(:,1)>0;as=sortrows(at,2);np=sum(at(:,2));
n=length (A);
```

```
» N=as(1:n-np,1).'
N =
   -7   -23     0
» P=as(n-np+1:n,1).'
P =
    1     2     8     4    56     3     9
» length(P)
ans =
    7
» length(N)
ans =
    3
» n
n =
   10
» np
np =
    7
»
```

3-мысал. *A* жиымы берілген. Жол мен бағанның қиылысқан жерінде максималдық мән орналасқан элементін өшіру керек.

```
» A=[1 2 -7; 8 4 -23; 56 0 3 ]
A =
    1     2    -7
    8     4   -23
   56     0     3
» maxa=max(max(A))
maxa =
   56
» [i,j]=find(A==maxa)
i =
    3
j =
    1
» A(i,:)=[];  A(:,j)=[];
» A
```

```
A =
    2   -7
    4   -23
»
```

4-мысал. *A* жиымы берілген. Матрицаның бірінші жолына сәйкес келетін жолдар санын анықтау керек.

```
» A=[1 2 -7; 8 4 -23;1 2 -7; 56 0 3;1 2 -7; ]
A =
    1    2   -7
    8    4   -23
    1    2   -7
   56    0    3
    1    2   -7
» v=size(A)
v =
    5    3
» n=v(1)
n =
    5
» af=repmat(A(1,:),n-1,1)
af =
    1    2   -7
    1    2   -7
    1    2   -7
    1    2   -7
» y=A(2:end,:)==af
y =
    0    0    0
    1    1    1
    0    0    0
    1    1    1
» ns=sum(all(y,2))
ns =
    2
»
```

Бақылау жұмысы. Векторлармен және матрицалармен жұмыс

Жұмыстың мақсаты:

1. MATLAB пакетінің негізгі жағдайларымен танысу.

2. MATLAB-та жиымдарды енгізу-шығару тәртібін және матрицалармен операцияларын игеру.

3. Диалогтық режимде жұмыс істеудің тәжірибелік жолдарын игеру.

Жұмыстың тапсырмасы

1 есеп. MATLAB интерфейсімен танысу.

2 есеп. MATLAB демонстрациялық мысалдарымен танысу.

3 есеп. Калькулятор режимінде келесі іс-әрекеттерді істеу керек:

- Бастапқы операндтарды енгізу.
- 1 және 2 операндтармен 1-ші операцияны істеу.
- Нәтижемен және 1 операндпен 2 операцияны істеу.
- Нәтижемен және 2 операндпен 3 операцияны істеу.
- 1 операндтың әр мүшесімен 3-тік дәрежесін орындау.

Бақылау сұрақтары

1. Бірөлшемді және көпөлшемді массивтерді енгізу мен шығару қалай жүзеге асады?

2. Векторлар мен матрицалар қалай беріледі?

3. Массивтің элементіне қаратпа.

4. «Қос нүкте» операторын қолдану.

5. Массивтің жолы мен бағанын жою.

6. zeros(n,m), ones(n,m) және eye(n,m) функциялары не үшін қолданылады?

7. rand(n,m), magic(n), diag(a) және fliplr(A) функциялары не үшін қолданылады?

8. rot90(a), inv(a), det(a) және size(A) функциялары не үшін қолданылады?

9. length(v), sum(a), prod(a) және sort(A) функциялары не үшін қолданылады?

10. sortrows(a), min(a)(max(a)), any(a) және all(A) функциялары не үшін қолданылады?

Тапсырма варианты

№	Операнд 1	Операнд 2	Операторы		
			1	2	3
1	V=[12 34 45; 54 61 45; 11 15 65]	v=[14 41 54; 62 90 45; 18 77 42]	*	./	+
2	V=[80 67 12; 34 11 41; 71 45 15]	v=[24 34 17; 21 87 45; 25 88 12]	/	.*	-
3	V=[19 77 62; 45 11 65; 12 67 26]	v=[65 18 90; 35 15 45; 13 46 32]	+	.\	/
4	V=[11 98 21; 67 45 32; 22 12 95]	v=[95 56 22; 45 21 45; 11 54 42]	-	.*	/
5	V=[67 34 23; 67 45 75; 62 56 43]	v=[10 34 22; 10 41 45; 17 12 77]	+	.\	*
6	V=[18 36 82; 45 32 45; 41 14 75]	v=[12 54 34; 61 45 41; 11 62 15]	/	./	-
7	V=[55 43 42; 18 28 45; 41 23 54]	v=[80 67 71; 31 34 11; 51 45 13]	/	.*	/
8	V=[32 28 19; 55 45 25; 34 12 36]	v=[19 77 41; 45 32 11; 14 67 71]	*	-	/
9	V=[14 34 16; 33 45 21; 41 15 21]	v=[11 98 55; 67 22 45; 22 14 15]	*	+	+
10	V=[15 23 46; 17 48 45; 59 31 97]	v=[67 34 33; 12 67 45; 13 56 14]	/	-	*
11	V=[10 34 18; 10 45 49; 27 15 23]	v=[18 36 71; 45 15 45; 19 14 91]	*	-	*
12	V=[95 56 84; 25 31 45; 54 12 45]	v=[55 43 20; 38 18 45; 15 23 28]	+	./	+
13	V=[18 90 71; 35 45 55; 33 46 62]	v=[32 28 10; 55 24 45; 63 34 37]	*	.*	-
14	V=[24 34 13; 87 45 47; 88 17 25]	v=[80 67 14; 11 34 11; 11 45 67]	/	.*	/
15	V=[14 41 67; 90 45 51; 77 16 42]	v=[19 77 12; 23 45 11; 14 67 29]	/	+	+

3. М-ФАЙЛДАР: СЦЕНАРИЙЛЕР ЖӘНЕ ФУНКЦИЯЛАР

MATLAB-та мәтіндік файлда жазылған командалардың ретпен орындалу режимі сияқты командалық режимде де жұмыс істеуге болады (командалар мен операторлар командалық терезеге енгізіледі, дереу орындалады және нәтижесі алынады). MATLAB-тағы команда мен операторлары бар файл .m кеңейтілуімен жазылады және ол М-файл деп аталады. М-файлдың екі түрі бар: сценарий және функция. Сценарийлер мен функциялар MATLAB-тың кез келген командалары мен операторларынан тұруы мүмкін.

М-файлды құру үшін File/New/M-file (құрал-саймандар жақтауындағы  New M-file батырмасы) командасын қолдану қажет. Бұл команда бағдарламаны енгізу мен шығаруға арналған М-файлдың редакторлық терезесін ашады. Енгізу аяқталғаннан кейін М-файл сақталған болуы тиіс (ағымдағы немесе кез келген басқа каталогта да). М-файлды қолдану үшін осы файлға алып баратын іздеу жолы орнатылуы тиіс.

Егер құрылған М-файлға өзгерістер енгізу талап етілсе, онда оны ашып, өзгеріс енгізіп және модификацияланған файлды сақтау қажет (File/Save командасы (File/Save as – басқа атпен сақтау) немесе редактордың құрал-саймандар жақтауындағы ұқсас батырма).

М-файлда бір жолда таңбалармен бөлінген бірнеше команда жазылуы мүмкін (нүктелі үтір), бірақ көрнектілік үшін оларды жеке жолдарда орналастырған ыңғайлы. М-файлдармен жұмыс істеген кезде нүктелі үтір жеке команда нәтижелерінің командалық терезесіне шығарылуы үшін де қолданылуы мүмкін. Ұзын өрнек бірнеше жолды алу мүмкін. Жолдарды тасымалдау кезінде «көп нүкте» таңбасы қолданылады. М-файлда айнымалылардың түрлері жарияланбайды, ал олардың мәндері меншіктелгеннен кейін автоматты түрде анықталады.

Бағдарламаны қолдану кезінде түсініктемелерді қолдануға кеңес беріледі, бағдарламаның орындалуы кезінде бұл түсініктемелер еленбейді. Түсініктеме – бұл пайыз (%) таңбасынан кейін орналасқан жол немесе оның бөлігі. Түсініктеме жеке

жолдан да тұруы немесе операторлардан кейін де орналасуы мүмкін.

Бірінші операторға немесе бірінші бос жолға тиісті түсініктемелер анықтамалық жүйе ретінде қолданылады. Түсініктеменің бірінші жолы келесідей командамен шығарылады: *lookfor* түйінді сөз және *help* каталог атауы. Ал *help* функция атауы командасы бос жолда немесе бірінші операторда орналасқан түсініктеменің барлық жолдарын экранға шығарады.

3.1. Сценарий

Сценарий көп ретті орындауды қажет ететін жұмыстарды автоматтандыру үшін арналған, яғни оларды қолдану командалық жолға команданы қайтара енгізу қажеттілігінен құтқарады. Сценарийдің кіріс және шығыс аргументтері болмайды. Сценарийдің орындалуы кезінде оның командасына жұмыс аймағындағы деректер қолжетімдік дәрежеде болады. Егер сценарийдің орындалуы кезінде айнымалылар құрылса, олар жұмыс аймағына орналастырылады және сценарийді орындау аяқталғаннан кейін де сонда қалады. Осылайша, сценарий тек жұмыс кеңістігінің айнымалыларымен ғана жұмыс істейді және өзінің жеке айнымалылары болмайды. Сондықтан да, бірнеше сценарийлердің орындалуы кезінде олардың ортақ айнымалыларының сәйкестігіне назар аудару керек.

Сценарийді орындау үшін командалық жолда сақталған файлдың атын енгізіп және Enter-ді басу немесе *run* файлдың аты командасын қолдану керек. Мысалы, *ex1.m* файлында сақталған бағдарламаны қосу үшін командалық жолға *ex1* немесе *run ex1* командасын енгізу қажет. Сондай-ақ бағдарлама редактор/түзеткіш терезесінде де іске қосылуы мүмкін.

М-файлды командалық жолдан шақырған кезде келесідей іс-қимылдар жүзеге асырылады:

1. Жұмыс кеңістігінде аттас айнымалының бар-жоғы тексеріледі.

2. Егер жұмыс кеңістігінде айнымалы болмаса, берілген атпен аттас функцияны іздеу орындалады.

3. Егер алдыңғы адымдар оң нәтиже бермесе, онда енгізілген атпен М-файлды іздеу орындалады: алдымен іздеу ағымдағы жұмыс каталогында, содан соң іздеу жолында орнатылған каталогтарда орындалады.

4. Егер файл табылса, ол орындалады. Ал егер ол орындалмаса, онда қате туралы хабарлама шығады.

Мысал. Бастапқы матрицаның бірінші бағанының максималдық және минималдық элементтерінің өзара қайта құрылуы үшін сценарий құру керек.

Сценарий мәтіні:

```
a=[3 4 5;6 4 2;9 7 5] % алғашқы матрица
[maxa,imax]=max(a(:,1)); % бірінші бағанның максималдық элементін анықтаймыз
[mina,imin]=min(a(:,1)); % бірінші бағанның минималдық элементін анықтаймыз
% өзара қайта құруды орындаймыз
% бірінші бағанның максималдық және минималдық элементі
v= a(imax,1);
a(imax,1)=a(imin,1);
a(imin,1)=v;
a  % қайта құрудан кейінгі матрица
```

3.2. Іздеу жолының қондырғысы

Орындауға тиісті болатын кез келген файл ағымдағы немесе іздеу (ену) жолының тізімінде көрсетілген каталогта орналасуы керек. Әйтпесе, талап етілетін файл табылмайды. Сондықтан да, егер файл іздеу жолы тізімінде көрсетілмеген каталогта сақталған болса, онда осы каталогқа жолды іздеу жолының тізіміне енгізу қажет. Келесі командалар іздеу жолының тізімімен жұмыс істеуге арналған:

Path - іздеу жолының тізімін экранға шығару,

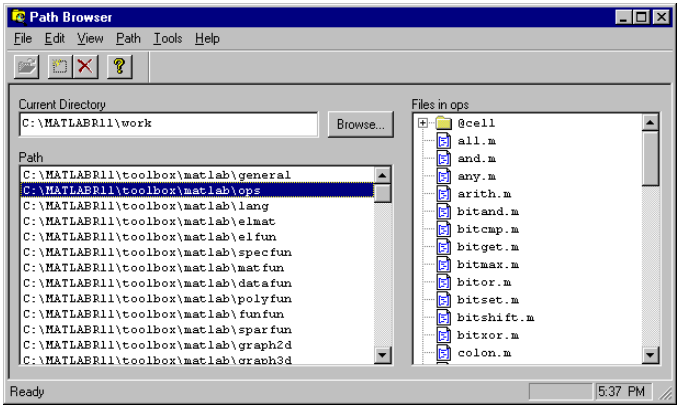
addpath іздеу жолы - каталогты іздеу жолы тізіміне енгізу, мысалы: addpath 'c:\matlab_w',

rmpath іздеу жолы - Каталогты іздеу жолы тізімінен өшіру, мысалы: rmpath 'c:\matlab_w'.

Айтылған командалардан басқа да іздеу жолын қараудың ыңғайлырақ құралы – Path Browser бар, бұл терезе File/ Set Path… командасымен немесе 3.1-суретте келтірілгендей командалық терезенің  Path Browser батырмасымен шақырылады.

Path Browser іздеу жолын қарауға, өшіруге және тағайындауға, олардың тізімдегі реттілігін өзгертуге, MATLAB-тың барлық файлын қарауға мүмкіндік береді.

Current Directory жолы ағымдағы каталогты бейнелеу және өзгертуге арналған (жолдың оң жағындағы Browse батырмасы).



3.1-сурет. Path Browser терезесі

Path жолы ену жолы тізімінен тұрады. *Files in* *Path* жолында белгіленген каталогтың аты жолы файлдар тізімі мен белгіленген каталогтың ішкі каталогтарынан тұрады. Path *Browser* мәзірінің командасы мен оларға сәйкес келетін құрал-саймандар жақтауындағы батырмалар 3.1-кестеде келтірілген.

3.1-кесте

Path Browser мәзірінің командалары

Мәзір	Команда	Құрал-саймандар жақтауының батырмалары	Қызметі
File	Open		Файлды ашу
	Save Path		Келесі жұмыс сеансы үшін ену жолын сақтау
	Exit Path Brouser		Path Brouser-ден шығу
Edit	Undo		Соңғы амалды болдырмау
	Redo		Соңғы амалды қайтару
View	Toolbar		Құрал-саймандар жақтауын шығару
	Status Bar		Күй тақтасын шығару
	Editor Debug-ger		М-файл редакторының терезесін ашу
	Path Browser		Path Browser-ды көрсету
	Workspace Browser		Workspace Browser терезесін ашу
Path	Add to Path		Каталогты іздеу жолы тізіміне қосу
	Remove from Path		Каталогты іздеу жолы тізімінен алып тастау
	Refresh		Іздеу жолын жаңарту (командалық терезедегі командамен өзгерту енгізгеннен кейін)
	Restore De-faults		Үнсіздікпен қабылданып қойылғандарды қалпына келтіру
Tools	Run		М-файлды орындау
	Customize		Күйге келтірулерді таңдау
	Options		Қызметтерді орнату
	Font		Әріпті орнату
Help	About Matlab Path Browser		Path Browser туралы ақпараттан қорытынды

Path аймағында каталогты басқа іздеу жолы тізімі позициясына орын ауыстыру үшін оны тышқанның сол жақ батырмасымен ұстап тұрып жылжыту қажет.

3.3. Функциялар

Функциялар белгілі бір тапсырманы шешетін бағдарламаның оқшауланған фрагментін іске асыру үшін арналған. Олар кіріс және шығыс аргументтерінен тұруы мүмкін (параметрлері бар және бір немесе бірнеше мәндерді қайтара алады). Функция параметрлері мен қайтарылу мәндері әр типті, әр түрлі жиымдар болуы мүмкін. Сценарий сияқты функциялар да MATLAB командалары мен операторларынан тұрады және. m кеңейтілуі бар мәтіндік файлдарда сақталады. Функция сақталатын М-файлдың атауы функция атауымен сәйкес келу керек. Функция MATLAB-қа ену тізіміне кіретін каталогтардың бірінде орналасуы тиіс.

Функция командалық жолдан, басқа функциялардан немесе сценарийлерден шақырылуы мүмкін. Егер функция қайтаратын шаманың типі бір өрнекке сәйкес болатындай болса, оның шақырылуы сол өрнекке енуі мүмкін.

Функция мәтіні (оның анықтамасы) функцияның тақырыбынан басталады, одан кейін функцияның денесі болады. Тақырып функцияның интерфейсін анықтайды және келесідей болады: function [*y1, y2, y3,* …]= функция атауы
(*x1, x2, x3*, …), мұндағы *y1, y2, y3*, … - қайтарылатын мәндер; *x1, x2, x3*, … - функция параметрлері.

Мысалы, бір а параметрі мен үш *x, y, z* қайтарылатын мәндері бар F1 атты функцияның мәні келесідей түрге ие: function [*x,y,z*]=*F1(a*).

a, b екі параметрлі және х бір қайтарылатын шамасы бар *F2* функциясының тақырыбы мына түрде көрініс табады: function x=F2(a,b).

a,b,c үш параметрі бар және ешқандай қайтарылатын шамасы жоқ F3 функциясы: function *F3(a,b,c).*

F4 функциясының параметрлері де, қайтарылатын мәндері де жоқ болса, онда function *F4.*

Функция MATLAB операторлары мен командалардан тұрады. Функцияда қайтару айнымалыларының меншікті мәндері болуы тиіс (кері жағдайда, функцияны шақыртқан кезде сәйкес хабарлама шығады).

Бірінші бос жолға дейінгі немесе бірінші операторға дейінгі функцияның тақырыбынан кейінгі орналасқан түсініктеме *help* функция атауы командасымен шығады. Осы түсініктеменің бірінші жолы сценарий түсініктемесінің бірінші жолында қолданылады.

Бірнеше мәнді (*Y1, Y2, Y3*, … айнымалыларына меншіктелетін) қайтаратын функцияны шақырту келесідей түрде жазылады: [*Y1, Y2, Y3*, …]=функция атауы (параметрлер тізімі үтір арқылы). Мысалы, тақырып алдында қарастырылған *F1, F2, F3, F4* функцияларының шақырылуын жазу керек болсын. Осы кезде қайтатын мәнге ие болатын айнымалылар атауы мен параметрлерінің мәнін көрнекілік үшін бас әріптермен жазылады:

```
[X,Y,Z]=F1(A)
X=F2(A,B) немесе F2(A,B) – өрнекте
F3(A,B,C)
F4
```

Мысал. Үлкен сандармен берілген матрица элементтерінің санын анықтау үшін функция құру. Осы элементтердің индекстерін анықтау. Функцияны анықтау мына түрде болады (l3_p2.m файлында сақталған):

```
function [k, ind]=fk(A,Y)
%Y үлкен сандармен берілген A матрицасының
элементтерінің санын анықтау
k=sum(sum(A>Y));%санды анықтау
[i,j]=find(A>Y);%, Y үлкен сандармен берілген A матрица
элементтерінің индексі
ind=[i,j];%индексті матрицаға жазамыз (бірінші баған -
жол, ал екінші- баған нөмірі)
```

А матрицасын құрамыз да командалық жолдан функцияны шақырамыз:

```
» A=[6 8 2;5 9 12;4 2 78]
A =
    6    8    2
    5    9   12
    4    2   78
» [g,h]=l3_p2(a,7)
g =
    4
h =
    1    2
    2    2
    2    3
    3    3
»
```

М-файлда бірнеше функцияның анықтамасы болуы мүмкін. Осы функциялардың біреуі негізгі файлдағыдай атқа ие болуы керек. Осы кезде (анықтамасы басқа файлда орналасқан сценарий немесе функцияның командалық жолынан) тек негізгі функция шақырылуы мүмкін. Қалған функциялар (олар көмекші немесе ішкі функциялар деп аталады) негізгісімен немесе бір-бірімен шақырылуы мүмкін.

Мысал. Кесінді ұзындығы дюйм (1дюйм=2,54см) арқылы берілген. Ұзындық мәнін метрлік жүйеге, яғни оны метр, сантиметр, миллиметрге ауыстыру керек. Мысалы, 21дюйм=0м 53см 3,4мм.

Есепті шешу үшін үш функция құрамыз: pr атты негізгі және екі көмекші D (дюймді миллиметрге ауыстыруға) және *M* (миллиметрді метр, сантиметрге ауыстыру үшін). М-файлдың pr.m тексті pr, *D* және *M* функцияларымен:

```
function [m, sm, mm]=pr(d)
% pr- дюймдағы кесінді ұзындығын метр, сантиметр,
миллиметрге ауыстыру
mm=D(d);
[m,sm,mm]=M(mm);
function mm=D(d)
```

```
% дюймді миллиметрге ауыстыру
mm=d*25.4;
function [m,sm,mm]=M(mm)
% миллиметрлерді метр, сантиметр, миллиметрге
ауыстыру
m=fix(mm/1000);% метрлер
mm=rem(mm,1000);% метрдің бүтін санын алып қаламыз
sm=fix(mm/10);% сантиметрлер
mm=rem(mm,10);% миллиметрлер
```

Суреттеу үшін ұзындығы 121 дюйм кесіндіні метр, сантиметр, миллиметрге түрлендіреміз (*pr* функциясының шақырылуына назар аударыңыз):

```
» [m1, sm1,mm1]=pr(121)
m1 =
    3
sm1 =
    7
mm1 =
   3.4000
» whos
  Name      Size         Bytes  Class
   m1         1x1             8  double array
  mm1       1x1             8  double array
  sm1       1x1             8  double array
Grand total is 3 elements using 24 bytes
» help pr
 Pr - ұзындығы дюйм кесіндіні метр, сантиметр,
миллиметрге айналдыру
» D(12)
??? Undefined function or variable 'D'.
»
```

D көмекші функциясын шақырған кезде командалық жолдан қате туралы хабарлама шығады.

Функцияда қолданылған барлық айнымалылар жергілікті болып табылады (жұмыс кеңістігінде және басқа да функциялар

үшін олар қол жетімсіз). Бұл алдыңғы мысалдағы whos командасының орындалу нәтижесін суреттейді. Бұл мысалдағы бір атқа ие әртүрлі функция айнымалылары негізінен әртүрлі айнымалылар болып табылады. Егер функцияны айтылған каталог функциясымен ғана шақыру мүмкіндігін шектеу қажет болса, онда private деп аталатын ішкі каталог құрылуы тиіс және файлды функциямен бірге сонда қою керек. Private каталогындағы барлық функциялар тек алдыңғы каталогтағы функциялармен ғана көрінеді. Private каталогын File/Set Path… командасының іздеу жолы тізіміне енгізу мүмкін емес.

Функцияны шақырған кезде айнымалыны іздеу сол атпен немесе келесі реттегі функциямен (сондай атпен алғашқы кездескен функциямен):

1. Құрылған функция.
2. Сол М-файлдағы ішкі функция.
3. Private ішкі каталогтағы М-функциясы.
4. Ағымдық жұмыс каталогындағы М-функция.
5. Іздеу жолында орнатылған каталогтағы М-функция (тізімдегі жоғарыдан төмен қарай ретпен).

Функцияны тапқан жағдайда ол орындалады, ал кері жағдайда қате туралы хабарлама шығады.

3.4. Аумақты айнымалылар

Функцияда жергілікті айнымалылар, ал сценарийде – жұмыс кеңістігінің айнымалылары қолданылады. Қажеттілік жағдайда жұмыс кеңістігінің аттас айнымалылары және бірнеше функциялар global айнымалылардың аттары үтір арқылы беріледі командасының көмегімен аумақтық түрде хабарландырылуы мүмкін. Бұндай хабарлама аумақтық айнымалылар қолданылатын кез келген функцияға қойылуы тиіс, егер аумақтық айнымалы жұмыс аймағында да қолжетімдік дәрежеде болуы үшін командалық терезеде жасалуы тиіс. Мысалы, *n* аумақтық айнымалысы *r1* және *r2* функцияларын шақырудың жалпы санын санағышы ретінде қолданылсын. Онда функцияны анықтау:

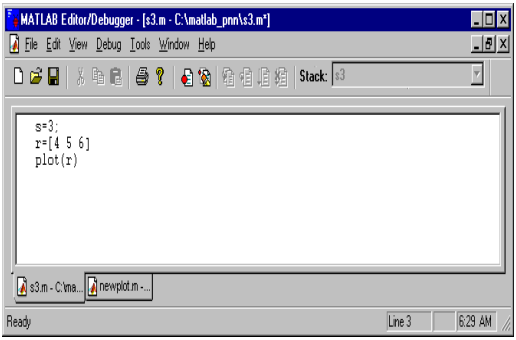
```
function r1
global n
n=n+1;
function r2
global n
n=n+1;
```

сессия фрагменті:

```
» global n
» n=0;
» r1
» n
n =
    1
» r2
» n
n =
    2
» r2
» n
n =
    3
» whos
  Name      Size          Bytes  Class
  n         1x1               8  double array (global)
Grand total is 1 elements using 8 bytes
»
```

3.5. М-файлдың редактор/дұрыстауы. Бағдарламаны дұрыстау

М-файлдың Editor/Dubugger редактор/дұрыстауы (3.2-сурет) М-файлды құру, түзету және дұрыстау үшін арналған. Редактор/дұрыстау командалық жолдан edit командасы немесе File/New/M-file (New M-file батырмасымен) командасымен шақырылуы мүмкін.



3.2-сурет. М-файлдың Editor/Dubugger терезесі

Егер редакторда бірнеше файл ашық тұрса, онда ол көптерезелі болады. Әрбір файл жеке қосымшаларда орналасады, олардың арасын айырып-қосу қосымшаның түбін тышқанмен түрту арқылы жүзеге асырылады. М-файл редакторының тақырып жолында белсенді файлдың атауы мен оған дисктегі жолы көрінеді. Егер аттан кейін «*» таңбасы бейнеленсе, онда бұл файл өзгеріс енгізілгеннен кейін сақталмағанын білдіреді.

File мәзірі М-файлмен жұмыс істейтін командалардан тұрады (құру, ашу, жабу, сақтау, басып шығару).

Edit мәзірінде М-файл тексті фрагменттерін орындалған іс-қимылдан бас тарту, оны қайталау, белгілеу, көшіру, кесу және қою командалары бар. Edit/Go To Line командасы берілген нөмір бойынша жедел түрде жолға өтуге мүмкіндік береді (жол нөмірі күй жолының оң жақ төменгі бұрышында бейнеленеді).

View мәзірі командасы құрал-саймандар жақтауының (Toolbar) және күй жолының (Status Bar) бейнеленуін басқаруға мүмкіндік береді; Path Browser және Workspace Browser шақыру командасынан тұрады. Evaluate Selection командасы М-файлдың белгіленген фрагментін орындау үшін арналған. Auto Indent Selection командасы автоматты орындалулар үшін қызмет атқарады.

Debug мәзірінің командалары және оларға сай келетін құрал-саймандар жақтауындағы батырмалар бағдарламаны дұрыстау үшін арналған. Олар 3.2-кестеде келтірілген.

3.2-кесте

Debug мәзірі командалары

Команда	Батырма	Сипаттама
Continue		М-файлды орындауды ол аяқталғанша немесе бірінші тоқтау нүктесіне дейін жалғастыру
Single Step		Ағымдағы жолды орындау
Step in		Ағымдағы жолды орындау, бірақ егер онда М-функциясының шақыруы болса, шақырылатын функцияның бірінші орындаушы жолына өтуін жүзеге асырады
Quit Debugging		Дұрыстау режимінен шығу
Set/Clear Breakpoint		Курсор орналасқан жердегі жолға бақылау нүктесін орнату/өшіру
Clear All Breakpoints		Барлық бақылау нүктелерін өшіру
Stop if Error		Қате бар болған жағдайда тоқтату
Stop if Warning		Ескерту болған жағдайда тоқтату
Stop if NaN or Inf		Егер нәтиже NaN немесе Inf болған кезде тоқтату

Tools мәзірі редактор, әріп параметрлерін реттке келтіру командаларынан, Run – М-файлды орындау командасынан тұрады.

М-файлды дұрыстау үшін тоқтау нүктесі мен бағдарламаның адымдық орындалуы қолданылады (3.2-кесте). Тоқтау нүктесі көрсетілген жолдар сол жағынан қызыл домалақпен белгіленеді. Тоқтау нүктесі берілгеннен кейін функция (сценарий) Tools/Run командасының орындалуына жіберіледі.

Бағдарламаны орындау кезінде тағайындалған бірінші тоқтау нүктесінде оның тоқтауы жүзеге асырылады. Бағдарламаның орындалуы тоқтау нүктесінде тоқтатылса, онда жолдың сол

жағында сары бағдар пайда болады. Оңға бағытталған бағдар бұл жолдың келесі ретте орындалатынын білдіреді, ал төменге қарай бағытталған бағдар функцияның соңына жеткендігін білдіреді. Тоқтау нүктесінде тоқтаған кезде командалық жолда командаға енуге шақыру таңбасы >>-тан К>>-ға өзгереді. Тоқтау нүктесінде айнымалылар мәнін, функция, өрнек параметрлерін талдауға болады. Айнымалының мәнін көру үшін бағдарлама мәтініндегі оның атының жанына курсорды алып бару жеткілікті, осыдан кейін экранда ішінде айнымалының мәні бар сары төртбұрыш шығады. Дұрыстау үдерісінде айнымалы мәні сондайақ командалық терезеге шығуы мүмкін. Бұл үшін айнымалыны белгілеп және View/ Evaluate Selection командасын қолдану керек. Бұдан басқа, бағдарламаны орындар алдында уақытша таңбаны өшіруге болады.

Дұрыстау үдерісінде айнымалылардың мәні өзгеруі мүмкін. Бұл үшін өзгерту енгізу керек жерде тоқтау нүктесін белгілеп және үзіліс кезінде командалық терезеде талап етілген мәндегі айнымалыны меншіктеуі тиіс.

Бағдарламаны орындауды Continue, Single Step немесе Step in командаларының бірімен жалғастыруға болады.

Мәзір және құрал-саймандар жақтауының көмегімен енгізілетін редактор/дұрыстаушы дұрыстау командаларынан басқа командалық терезенің енгізу жолында енгізілетін ұқсас командалар да бар.

Бақылау сұрақтары

1. М-файлдарды жасау және қолдану жолдарын түсіндіріңіз.
2. М-файлдың қандай түрлерін білесіз?
3. Сценарийдің қызметін түсіндіріңіз.
4. Функцияның қызметін түсіндіріңіз.
5. Іздеу жолдары қалай қойылады?
6. Комментарийлер қалай беріледі?
7. Аумақты айнымалылар деген не?
8. Бағдарламаны қосу және түзеу қалай жүреді?

4. ГРАФИК ТҰРҒЫЗУ

MATLAB-та екі немесе үш өлшемді графиктерді тұрғызуға және қалыптауға болатын құралдар бар.

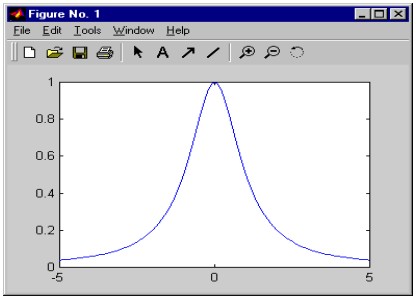
4.1. Екі өлшемдік графиктерді тұрғызу

Екі өлшемді графиктер декарттық және полярлық координаттар жүйесінде сызықтық және логарифмдік масштабтарда құрылуы мүмкін.

Функция графигін тұрғызу үшін аргумент мәнінің векторын беру қажет, функция мәнінің векторын есептеп және график функциясын шақыру керек.

4.1.1. Декарттық координат жүйесінде график тұрғызу

Сызықтық масштабта функция графигін тұрғызу үшін plot(X,Y) функциясы қолданылады. *X* векторында функцияның аргумент мәндерінің жиымы, ал *Y* векторында функция мәндерінің жиымы жазылуы керек. Plot функциясы график нүктелерін түзу сызықтармен біріктіреді. Мысалы, [-5,5] аралығында $f(x)=\frac{1}{x^2+1}$ функциясының графигін құру керек. Ол үшін командa жолында `»x=-5:0.1:5;y=1./(x.^2+1);plot(x,y)` теріледі. Enter-ді басқанда экранда графигі бар терезе пайда болады. Графикті шығарған терезеге график аймағы, құрал-саймандар жақтауы және мәзір кіреді. Графиктерді тұрғызу командасын қолданғанда график терезесі автоматты түрде ашылады (4.1-сурет).



4.1- сурет. $f(x)=\frac{1}{x^2+1}$ функциясының графигі

Егер plot(Y) функциясының аргументі ретінде бір бірөлшемді жиым болса, онда Y векторының элементтерінің индекстерінен тәуелділік құрылады. Егер *Y* векторының элементтері комплекстік болса, онда plot(real(Y),imag(Y)) графиктері құрылады. Егер *Y* векторы екі өлшемді жиым болса, онда баған элементтерінің жол нөмірінен тәуелділігі құрылады.

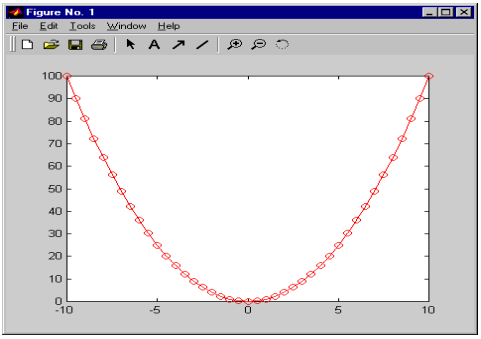
Егер plot(X,Y) командасының бір немесе екі аргументі екі өлшемді жиым болса, онда келесідей құрылымдар беріледі:

– егер *X* жиымы бір өлшемді, ал *Y* - екі өлшемді болса, онда *Y* жиымының бағандарының *X* векторының элементтерінен тәуелді графигі құрылады;

– егер *Y* жиымы бір өлшемді, ал *X* - екі өлшемді болса, онда *Y* векторының элементтерінің *X* жиымының бағандарынан тәуелді графигі құрылады;

– егер *X* және *Y* жиымдары екі өлшемді болса, онда *Y* жиымының бағаны *X* жиымының бағанынан тәуелді графигі құрылады.

Plot(X,Y,LS) командасы *LS* жолдық айнымалының көмегімен (дәйекшеге алынады) сызықтарды белгілеуге, түсін анықтауға және нүкте маркерін беруге мүмкіндік береді. Мысалы, *»a=-10:0.5:10;plot(a,a.^2,'ro-')* командасы [-10,10] аралығында дөңгелек маркермен және тұтас қызыл сызықпен параболаны салуды береді (4.2-сурет, 4.1-кесте).



4.2-сурет. Сызық қалыбын беру

4.1-кесте

Сызықтардың параметрлері

Сызықтың типі	символ	Сызық-тың түсі	символ	Маркер типі	символ
Біртекті	-	Жасыл	y	нүкте	.
үзік	--	күлгін	m	қосу	+
пунктирлі	:	көгілдір	c	жұлдызша	*
үзік-пунктирлі	-.	қызыл	r	дөңгелек	o
маркер бар болғандағы символ жоқ	сызық жоқ	Жасыл	g	крест	x
		Көк	b	квадрат	s
		Ақ	w	Ромб	d
		Қара	k	Бесбұрыш	p
				алтыбұрыш	h
				солға бағыттауыш	<
				оңға бағыттауыш	>
				төменге бағыттауыш	∨
				жоғарыға бағыттауыш	∧

Plot(X1,Y1,LS1, X2,Y2,LS2, …) командасы әртүрлі параметрлі сызықтармен графиктер тобын салуды орындайды.

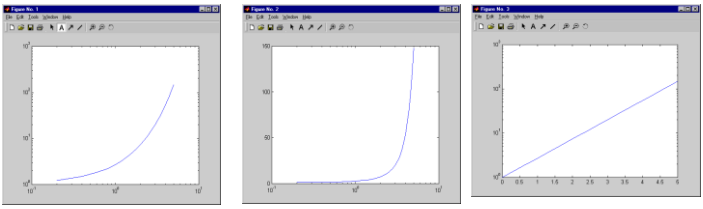
Plot(X,Y,LS, ‘PN’,PV …) командасы график тұрғызуда қолданылатын сызықтың қасиеттерін береді (‘PN’ – қасиеттің

атауы, *PV* – қасиеттің берілген мәні). Мысалы, алдыңғы мысалдағы сызықтың жуандығына 3 мәнін беру үшін берілетін команда *»a=-10:0.5:10;plot(a,a.^2,'r-h','LineWidth',3)* түрде болады.

Екі координаттық осьте логарифмдік масштабта график салу үшін plot функциясының орнына бұрынғы аргументтер жиынтығымен loglog функциясы қолданылады. *X* осі бойынша логарифмдік масштабта және *Y* осі бойынша сызықтық масштабта функция графигін тұрғызу semilogx функциясы арқылы орындалады. Ал semilogy функциясы Y осі бойынша логарифмдік масштабты және *X* осі бойынша сызықтық масштабта функция графигін тұрғызады. Мысалы:

```
»x=0:0.2:5;loglog(x,exp(x))
»figure;
»semilogx(x,exp(x))
»figure;
»semilogy(x,exp(x))
»
```

Нәтижелері 4.3–суретте көрсетілген.

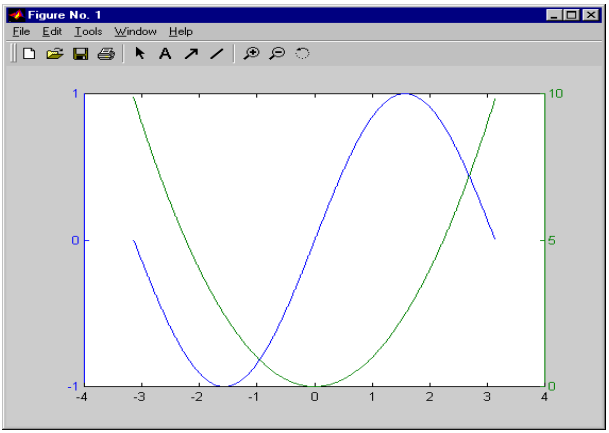


4.3-сурет. Логарифмдік масштабта тұрғызылған функция графигі

Plotyy(x1,y1,x2,y2) функциясы *x1, y1* және *x2, y2* жиымдарымен берілген екі координат осі бар екі функция графигін тұрғызады. (сол жағындағы ось бірінші графикке, ал оң жағында екінші графикке сәйкес келеді). Егер *y1, y2* – екі өлшемді жиым болса, графиктер саны ондағы бағандардың санына тең болады.

Мысалы, -π-ден π-ге дейінгі аралықта синусоида және парабола тұрғызайық:

»x=-pi:0.01:pi;plotyy(x,sin(x),x,x.^2). Нәтижелері 4.4-суретте көрсетілген.



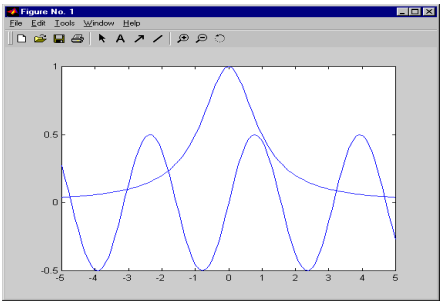
4.4-сурет. Екі координаттық осьтері бар графиктер

Графиктік терезенің құрал-саймандар жақтауындағы ,  және  пернелері батырмалары графиктік аймаққа мәтін, бағыттауыш және сызықтар қосу үшін арналған. График тұрғызылғаннан кейін графиктерді және оның құрама бөліктерін жөндеу, осьтер мен сызықтардың параметрлерін беру Edit және Tools командалары арқылы беріледі. Графиктік терезені толық сақтау үшін File/Save командалары қолданылады, ал қолданыстағы файл .fig кеңейтілуі арқылы жазылады. Графиктік терезені ашу File/Open командасымен іске асырылады.

Графиктік терезені белгілі бір графикалық қалыпта сақтау үшін оның кеңейтілімін көрсету керек. Графиктерді экспорттау үшін File/Save және File/Save *As* командаларынан басқа File/Export командасын қолдануға болады, алайда «Тип файла» тізімінде сақтау кезінде керекті қалыпты таңдау керек.

4.1.2. Бір терезеде бірнеше график тұрғызу

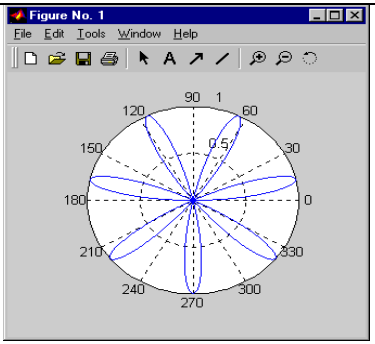
Графикті тұрғызудың кезекті командасын бергенде терезеде бұрынғысы жойылып, жаңа график пайда болады. Бір терезеде бірнеше график тұрғызу үшін hold on (алдыңғы графикті сақтау режимін іске қосады) командасы қолданылады. Бұрынғы графиктік сақтау режимін алып тастау үшін hold off командасы қолданылады (алдыңғы графикті сақтау режимін тоқтатады). Hold командасы бір режимнен екінші режимге өтуді жүзеге асырады. Мысалы: *»x=-5:0.1:5; y=1./(x.^2+1); plot(x,y); hold on; plot(x,0.5*sin(2*x)); hold off*. График тұрғызудың нәтижелері 4.5-суретте көрсетілген.



4.5-сурет. Hold on командасы арқылы бір терезеге бірнеше график тұрғызу

4.1.3. Полярлық координаттар жүйесінде график тұрғызу

Мәндері *F* және *R* жиымдарына жазылатын полярлық бұрыш пен полярлық радиус арқылы берілген функцияның графигін полярлық координаттар жүйесінде тұрғызу polar(F,R,LS) функциясын қолдану арқылы іске асырылады. Міндетті емес жолдық айнымалы *LS* сызықтың қасиетін береді (plot функциясындағыдай). Егер *F* және *R* жиымдары бірдей мөлшердегі екі өлшемді жиымдар болса, онда графиктер олардың әрқайсысының бағандары бойынша салынады. Мысалы, (0,π) аралығында ρ(φ)=sin(7φ) функциясының графигін тұрғызу керек.

*»x=0:0.001:pi;polar (x,sin (7*x))*	

4.6.1-сурет. ρ(φ)=sin(7φ) функциясының графигі

4.2. Графиктік терезелермен жұмыс

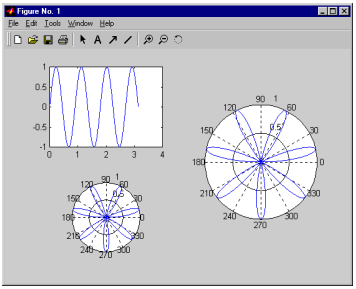
Графиктік терезеге өту үшін немесе жаңа графиктік терезе ашу үшін figure(n) командасы қолданылады. Мұндағы *n* графиктік терезенің нөмірі. Егер берілген нөмірдегі графиктік терезе жоқ болса, онда жаңа терезе пайда болады. Егер нөмірмен берілген терезе бар болса, қалған графиктік командалар сол терезеге қатысты болады. Параметрсіз figure командасы ең кіші нөмірдегі жаңа терезені ашады.

Close(n) немесе close *n* командасы *n* нөмірлі графиктік терезені жабады. Close all командасы барлық ашық тұрған графиктік терезелерді жабады.

4.3. Графиктік терезелердің шағын терезелерге бөлінуі

Графиктік терезені бірнеше терезелерге бөлуге болады және олардың әрқайсысына әр түрлі функциялардың графиктерін тұрғызуға болады. Терезені жалпы саны тігінен *n*, ал көлденеңінен m болатын шағын терезелерге бөлу үшін және нәтижені реті бойынша *k*-ші терезесіне шығару үшін (егер солдан оңға және жоғарыдан төмен қарай санасақ) subplot (*n, m, k*) функциясы қолданылады. Мысалы:

```
»x=0:0.001:pi;
y=sin(7*x);
»sub
plot(2,2,1);plot(x,y);
»sub
plot(2,2,3);polar(x,y);
»sub
plot(1,2,2);polar(x,y);
»
```

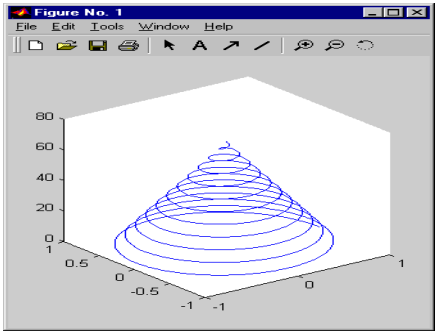


4.6.2-сурет. Графиктік терезені бірнеше терезелерге бөлу

4.4. Үш өлшемді графиктің тұрғызылуы

Кеңестіктегі сызықтарды құру үшін plot3(X,Y,Z) функциясы қолданылады. Бұл функция plot функциясының үшөлшемді аналогы болып табылады және *X, Y, Z* жиымдарында жазылған абсцисса, ордината және аппликата нүктелерін тік сызықтармен қосу үшін арналған. Ал келесі көрініс кеңістіктегі қисық сызықтың пайда болуына алып келеді (4.7– сурет):

```
»z=0:0.01:70; x=(1-z/70).*cos(z); y=(1-z/70).*sin(z); plot3(x,y,z).
```



4.7-сурет. Plot3 функциясын қолдану арқылы құрылған қисық

Екі айнымалысы бар функция графигін тұрғызу үшін:

1. [X,Y]=meshgrid (*x*,*y*) функциясының көмегімен абсциссасы *x* және ординатасы *y* болатын графиктің әрбір түйіндік нүктелеріндегі мәндер арқылы *X, Y* матрицалары құрылады. Сонымен бірге матрицаның әр *X* жолы *x* векторының көшірмесі, ал матрицаның әр *Y* бағаны *y* векторының көшірмесі болады.

2. Әрбір түйіндік нүктеде екі айнымалысы бар функция мәндерінің *Z* матрицасы есептелінеді.

3. Үш өлшемді графиктер тұрғызу функцияларының біреуі қолданылады (4.2-кесте).

4.2-кесте

Үш өлшемді графиктерді тұрғызу функциялары

Функция	Функцияның сипаты
mesh (x, y, z)	түйіндік нүктелердің координаттары x,y,z матрицаларының құрамында болатын беткі қабаттың қаңқасын тұрғызу
meshc (x, y, z)	XY жазықтығына проекцияланатын функцияның деңгей сызығымен шектелген беттің қаңқасын тұрғызу
meshz (x, y, z)	XY жазықтығына перпендикуляр проекцияланатын жазықтықтың шеткі нүктелерімен шектелген қаңқалы бетті тұрғызу
surf (x, y, z)	түйіндік нүктелердің координаттары x,y,z матрицаларының құрамында болатын боялған бетті тұрғызу
surfc (x, y, z)	XY жазықтығына проекцияланатын боялған бет пен функцияның деңгей сызықтарын тұрғызу
contour(x, y, z)	түйіндік нүктелердің координаттары x,y,z матрицаларының құрамында болатын беттік деңгей сызықтарын тұрғызу. Егер contour функциясы қайтаратын параметрді clabel функциясына берілсе, яғни clabel(contour(x, y, z)), онда графиктегі деңгей сызықтарына функция мәндері түседі.
contourf(x, y, z)	аралықтарын әр түрлі түстермен бояп беттің деңгей сызықтарын тұрғызу
contour3 (x, y, z)	кеңістіктегі беттің деңгей сызықтарын тұрғызу

Тұрғызылған графиктің сырт пішінін басқару үшін 4.3-кестеде келтірілгендей әр түрлі функциялар қолданылады.

4.3-кесте

Графиктің сырт пішінін басқару үшін арналған функциялар

Функция	Функцияның сипаты
hidden on (hidden off)	кестедегі көрінбейтін сызықтарды алып тастауды қосу (ажырату)
shading	беттерді көлеңкелеудің келесі тәсілдерінің бірін орнату: shading faceted - қара қырлы ұяшықтарының бір қалыпты боялуы; shading flat - ұяшықтар немесе қырлардың тор түйінінің түсімен тәуелділікте боялуы; shading interp - тор түйіндеріндегі түс интерполяциясымен анықталатын түспен боялуы
View	шолу нүктесін басқару: view(A,E) - E дөңестік бұрышын және A азимутын орнатады (азимут - Z осін айналдыра Y осінің теріс бағытында сағат тіліне қарсы айналдыру бұрышы, дөңестік бұрыш - координата басынан шыққан шолу нүктесіне дейінгі кесіндімен XY жазықтығы аралығындағы бұрыш); view ([x,y,z]) шолу нүктесін (x, y, z) координаталарымен береді; view(2), view(3) - шолу нүктесі үшін екі өлшемді және үш өлшемді графиктерінің кестелер бойымен қабылданған нүктелерінің орнын көрсетіп отырады

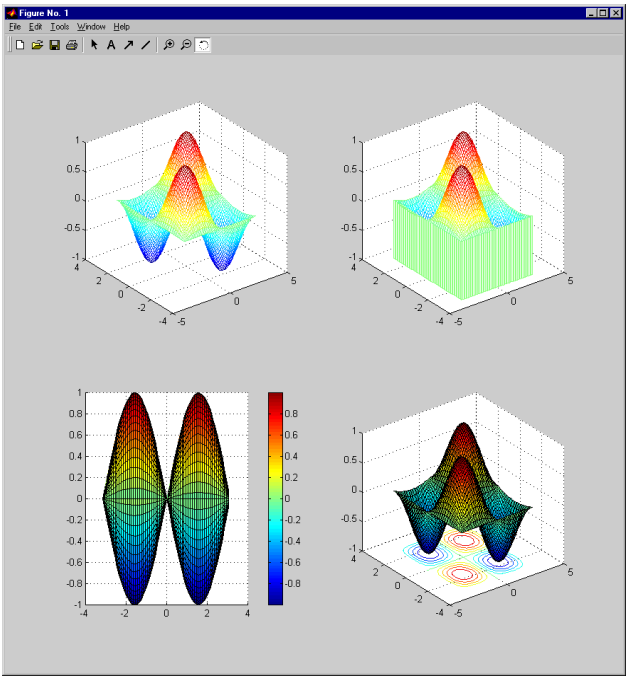
4.3-кестенің жалғасы

Colorbar	Z осінің шкаласы мен ағымдағы түс түрінің тік бағанын шығару
colormap(MN)	түс түрін MN атты түс түріне ауыстыру. Стандартты және қосымша түс түрлерінің атаулары: cool (көгілдір мен қарақошқыл рең), gray (сұр рең), hot (ақ-сары-сарғылт-қызыл-қара), hsv (кемпірқосақ түсі), jet (қызыл-сары-жасыл-көгілдір-көк), copper (мыс реңі), autumn (сары-сарғылт-қызыл), winter (көк-жасыл), summer (сары-жасыл), spring (сары-қарақошқыл), pink (қоңыр рең). Үнсіздікпен қолданылатын түс түріне қайтып келу colormap('default') командасы арқылы іске асырылады
Camlight	жарық көзін жасау: camlight (A,E) – берілген E дөңестік бұрыштық және A азимуттық координаталарында; camlight headlight – шолу нүктесінде; camlight right – шолу нүктесінен жоғары және оңға; camlight left – шолу нүктесінен жоғары және солға
Material	Беттік материалдың қасиетін беру: material shiny – жылтыр; material dull – бұлыңғыр; material metal – металл түстес; material default – үнсіздікпен қабылдану
brighten(B)	Түс түрлерінің айқындылығын өзгерту: 0<B<1 болғанда айқындылық артады, 1<B<0 болғанда айқындылық кемиді

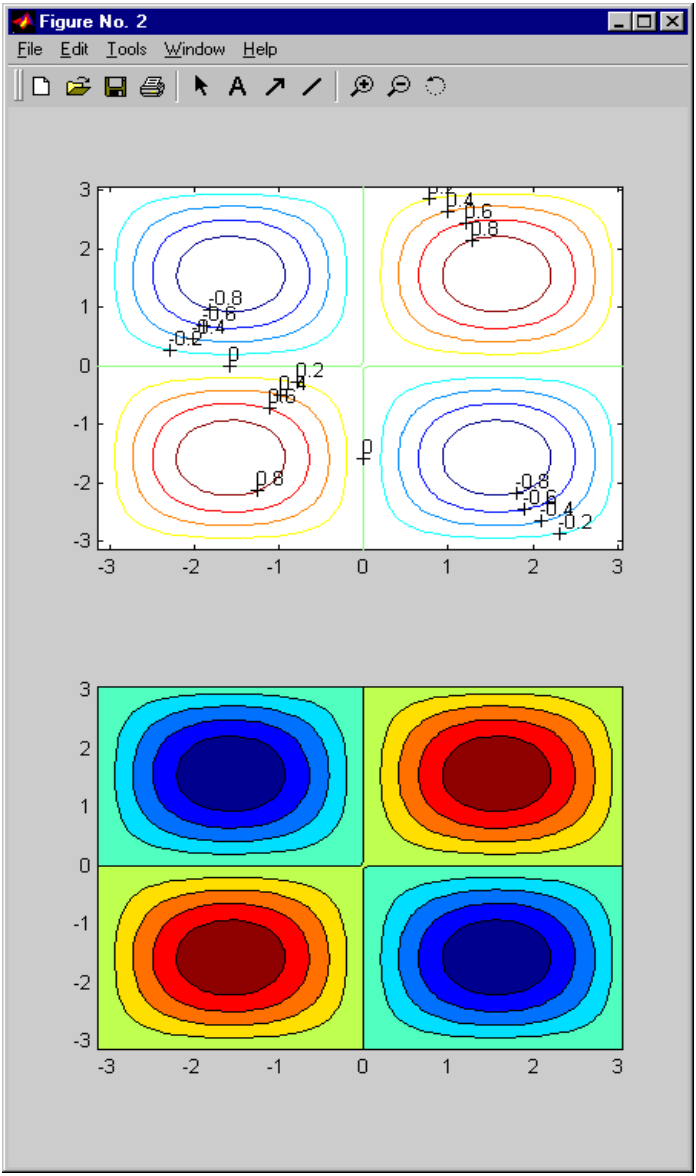
Үшөлшемдік график тұрғызудың мүмкіндіктерін екі айнымалысы бар $z(x,y)=\sin(x)*\sin(y)$ функциясының графигін салу арқылы көрсетуге болады. Ол үшін

```
»x=-pi:0.1:pi; y=x; [x1, y1]=meshgrid(x,y);
z=sin(x1).*sin(y1);
»subplot(2,2,1); mesh(x1, y1, z);
»subplot(2,2,2); meshz(x1, y1, z);
»subplot(2,2,3); surf(x1, y1, z); view ([0,-pi,0]); colorbar; view
([0,-pi,0])
»subplot(2,2,4); surfc(x1, y1, z);
»figure(2)
»subplot(2,1,1); clabel(contour(x1, y1, z));
»subplot(2,1,2); contourf(x1, y1, z);
»
```

командалары қолданылып, 4.8 және 4.9-суреттерде көрсетілген графиктер алынды.



4.8-сурет. $z(x, y) = \sin(x) * \sin(y)$ функциясының графиктері



4.9-сурет. $z(x,y) = \sin(x) * \sin(y)$ функциясының графиктері

4.5. Графиктік терезеге жазулар, тақырыптар және координаталық кесте шығару

График терезесінде координаталық кестені салу үшін grid on командасы қолданылады. Ал координаталық кестені алып тастау үшін grid off командасы қолданылады. Grid командасы бір күйден екінші күйге ауыстыру үшін қолданылады.

График аймағына жазу шығару үшін келесідей функциялар қолданылады (SV мәтіндік айнымалының дәйекшеге алынған түріндегі функцияны шақыруы берілсе):

xlabel(SV) – x осінің атауы;

ylabel(SV) – y осінің атауы;

zlabel(SV) – z осінің атауы;

title(SV) – графиктің үстіңгі жағына орналасқан тақырыбы;

text(X,Y,Z,SV) - (*X*,*Y*,*Z*) координаттық нүктеден басталатын мәтін. Нүкте координатасы сол график тұрғызылған координаттар жүйесінде беріледі. Жазықтықтағы график үшін *Z* координатасы болмайды;

gtext(SV) – SV жолының шығар орнын көрсететін белгіні шығару командасы. Енгізуді тоқтату үшін кез келген пернені басып немесе тышқанмен шерту керек.

legend(SV1,SV2,SV3 …pos) – График аймағына әр графикке, сызықтың немесе маркердің түрі және түсініктеме беретін мәтіндік SV1, SV2, SV3 айнымалылары түрінде жазылған түсініктемелік мәтіні бар әфсана легенданы қосады.

Pos параметрі арқылы әфсана легенданың орны беріледі:

Pos мәні	әфсана легенданың орны
-1	- графиктің оң жақ сырты
0	- графиктің нүктелерін әфсана легендамен неғұрлым азырақ жабатындай
1	- графиктің оң жақ жоғарғы бұрышы
2	- графиктің сол жақ жоғарғы бұрышы
3	- графиктің сол жақ төменгі бұрышы
4	- графиктің оң жақ төменгі бұрышы

Box +on (box off) командасы үшөлшемді объект орналасқан параллелепипедтің пішінін көрсету (алып тастау) үшін қолданылады. Box командасы бір күйден екінші күйге ауыстыру рөлін атқарады.

4.6. График масштабын өзгерту

Axis([xmin, xmax, ymin, ymax]) командасы екіөлшемді графиктің *x, y* координаттары бойынша, ал axis([xmin, xmax, ymin, ymax, zmin, zmax]) – үшөлшемді графиктің *x, y, z* осьтері бойынша масштабты орнатады. Xmin, ymin, zmin параметрлері *x, y, z* осі бойынша ең кіші мәндерді береді де, ал xmax, ymax, zmax, тиісінше, ең үлкен мәндерді береді. Axis командасы барлық осьтер бойынша [xmin, xmax, ymin, ymax, zmin, zmax]) шектік мәндері бар векторды параметрсіз кері қайтарады.

Axis('auto') командасы барлық осьтер бойынша, axis('auto x') командасы *x* осі бойынша, ал axis('auto xy') *x* және *y* осьтері бойынша шектік мәндердің автоматты түрде орнатылуын береді. Axis equal командасы барлық осьтер бойынша бірдей масштабтық коэффициент орнатады.

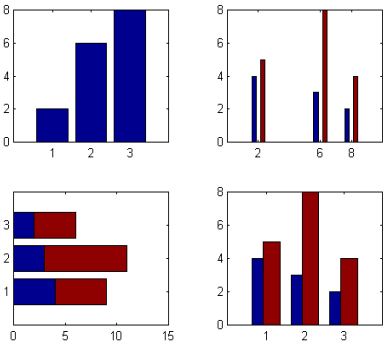
4.7. Арнайы график

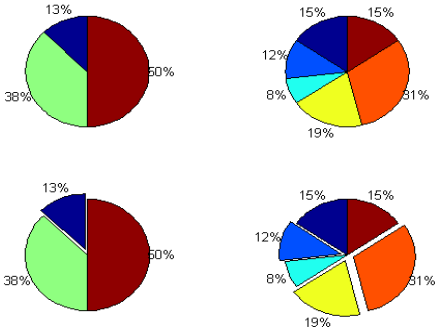
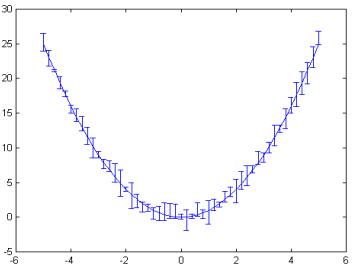
Арнайы екіөлшемді график функциялары 4.4-кестеде берілген.

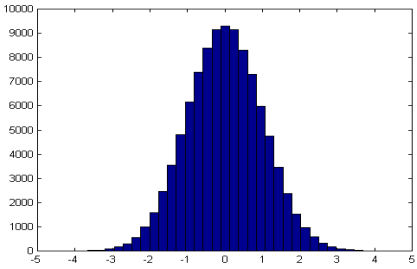
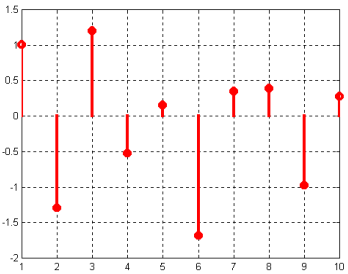
4.4-кесте

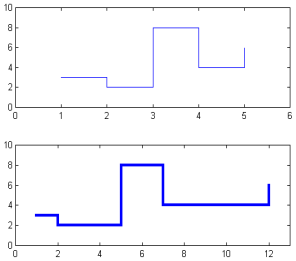
Арнайы екіөлшемді график

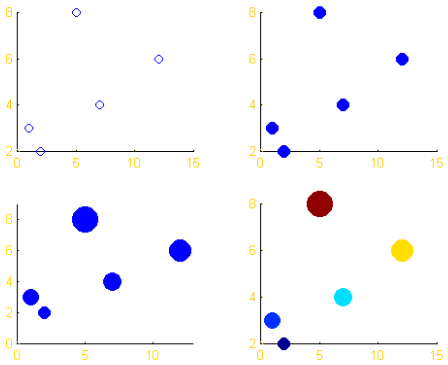
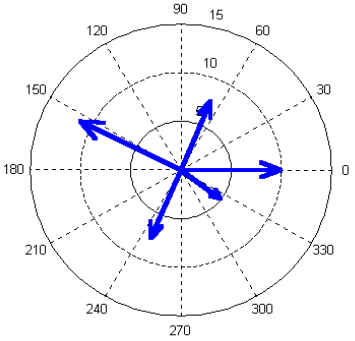
Функция	Сипаттамасы
bar (Y,h,s) bar (X,Y,h,s)	h баған биіктігінің салыстырмалы ені (үнсіздікпен h=0.8) мен s стиліндегі (баған ені мен стилі функцияны шақыру кезінде берілмеуі мүмкін) тік (bar) және көлденең (barh) бағандық диаграммалар тұрғызу: bar (Y,h,s) - егер Y бірөлшемді жиым болса, онда вектордың әр элементіне диаграмманың бағаны сәйкес келеді, егер Y – матрица болса, онда жол элементтерінің мәндеріне сәйкес болатын бағандар тобы шығады (ал x осі жол нөміріне сәйкес болатын белгілермен белгіленеді); bar (X,Y,h,s) – егер Y бірөлшемді жиым болса, онда X векторының элементтерінің мәндерінмен анықталатын орында Y векторының элементтері шығарылады, егер Y матрица болса, онда X век-

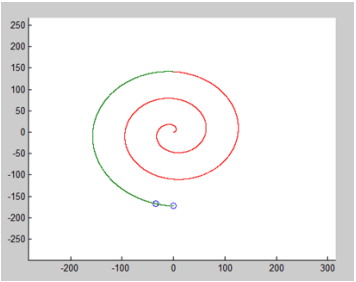
Функция	Сипаттамасы
	торының элементтерінің мәндерімен анықталатын орындарда Y матрицасының элементтеріне сәйкес бағандар тобы орналасады; мүмкін болатын стильдер: ‘group’ – диаграмма бағандар топтары түрінде (бұл стиль үнсіздікпен қолданылады), ‘stack’ – Y матрицасының әрбір жолына әр бөлігінің ұзындығы Y жиымының элементіне сәйкес келетін мәнмен анықталатын бір құрама баған сәйкес келеді. Мысалы: »x=[2 6 8];y=[4 5; 3 8; 2 4];subplot(2,2,1);bar(x);subplot(2,2,2); bar(x,y,0.5) »subplot(2,2,3);barh(y,'stack');subplot (2,2,4);bar(y,1.5) 
pie	дөңгелек диаграмма құру: pie(x) – x жиымының әрбір элементін дөңгелек диаграмманың секторы түрінде бейнелейді. pie(x,v) – қосымша дөңгелек диаграммада v жиымының нөлге тең емес элементтері сәйкес болатын қосымша секторлар бөлінеді (v жиымы x жиымы сияқты бір өлшемде болуы тиіс) Мысалы: »x=[2 6 8];y=[4 5;3 8;2];subplot(2,2,1);pie(x);subplot(2,2,2);pie(y) »c=[1 0 0];d=[0 1; 1 1; 0 0];subplot(2,2,3);pie(x,c);subplot (2,2,4);pie(y,d

Функция	Сипаттамасы
	
errorbar	графикті аралық қателігін көрсете тұрғызу. errorbar(y,e) және errorbar(x,y,e) функциялары графиктің әр нүктесіне қатысты ±e қателігімен тұрғызуға арналған errorbar(x,y,L,U) функциясы y-ің x-тен тәуелділігінің графигін әр нүктесіне қатысты –L және +U қателігімен тұрғызуға арналған Мысалы: »x=-5:0.2:5;y=x.^2;l=rand(1,length(y))*2;u=rand(1,length(y))*2; »errorbar(x,y,l,u) 
hist	гистограмма тұрғызу. hist(y) – 10 аралық үшін гистограмманы салады, hist(y,n) - n аралық үшін гистограмма салады. Мысалы:

Функция	Сипаттамасы
	»x=randn(100000,1); »hist(x,40) 
stem	stem(y,LS) - *y* бірөлшемді жиымының элементтерінің маркермен бітетін тік сызықтар түріндегі индекстен тәуелді графигін тұрғызу. stem(x,y,LS) - *y* бірөлшемді жиымының элементтерінің x векторының элементтерінен тәуелділігінің графигін тұрғызу. Мысалы: »x=randn(10,1);k=stem(x,'r');grid on;set(k,'LineWidth',3) 
stairs	stairs(Y,LS) – ординатасы бірөлшемді *Y* жиымы, ал абсциссасы *X* болатын сатылы график тұрғызады. Строковая переменная *LS* жолдық айнымалы сызық параметрлерін береді; stairs(X,Y,LS) - *X* жиымының элементтерінің мәндері сатылы графиктің абсциссасы, ал *Y* жиымының элемент мәндері ордината болады. Мысалы: »x=[1 2 5 7 12];y=[3 2 8 4 6];

Функция	Сипаттамасы
	»figure(2);subplot(2,1,1);stairs(y);axis([0,6,0,10]); »sub- plot(2,1,2);h=stairs(x,y);set(h,'LineWidth',3);axis([0,1 3,0,10]); 
scatter	абсцисса және ордината мәндері *X* және *Y* век-торларында болатын жазықтықта нүктелер тобын құру: scatter(X,Y) – көк түсті шеңбер түрінде; scatter(X,Y,S) – шеңбер өлшемдері *S* векторы ар-қылы беріледі; scatter(X,Y,S,C) – шеңбер түсі C векторы арқылы беріледі; scatter(… , 'filled') – шеңбер боялады. Мысалы: »x=[1 2 5 7 12];y=[3 2 8 4 6]; »figure; »subplot(2,2,1);scatter(x,y) »subplot(2,2,2);scatter(x,y,100,'filled') »subplot(2,2,4);scatter(x,y,50*y,y,'filled') »subplot(2,2,4);scatter(x,y,50*y,y,'filled') »subplot(2,2,3);scatter(x,y,50*y,'filled') »axis([0,13,0,9])

Функция	Сипаттамасы
	
compass	compass(*Z, LS*) - *Z* жиымына жазылған комплекстік сандарға геометриялық түсінік беретін векторлар салады; compass(X,Y, LS) – compass(X+j*Y)-ке сәйкес; Міндетті емес LS параметрі жолдық айнымалы болады және сызықтың параметрлерін береді. Мысалы: »z=[3+7j 4- 3j -10+5j -3-7j 10]; »h=compass(z);set(h,'LineWidth',3) 
comet	Нүкте қозғалысын кометаның басы мен құйрығы түріндегі траектория бойынша бейнелейді: comet(Y) - траектория *Y* бірөлшемді жиымымен берілген (*Y* жиымының элементтерінің индекстері нүктенің абсцисса мәндеріне, ал элемент мәндері ордината мәндеріне тең);

Функция	Сипаттамасы
	comet(X,Y) - траектория *X* және *Y* бірөлшемді жиымдармен берілген; comet(X,Y,P) – *P* параметрімен комета құйрығының салыстырмалы ұзындығы беріледі (комета құйрығының ұзындығы P*length (Y) - ге тең). Мысалы: Комета шиыршық бойынша қозғалады. »f=0:0.001:10*pi;r=f*10; x=r.*cos(f);y=r.*sin(f); »figure;comet(x,y)  comet3(X,Y,Z,P) функциясы *X, Y, Z* бірөлшемді жиымдармен берілген нүктенің кеңестіктегі траекториясы бойынша қозғалысын салады. *P* параметрі арқылы комета құйрығының ұзындығы беріледі

4.8. Қосымша мысалдар

1-мысал. Берілген [xn; xk] аралықта берілген

$$y(x)=\begin{cases} 50\dfrac{\sin(x)^2}{x} & \text{егер } x<-4 \\ \dfrac{10\cdot e^x}{x^2+1} & \text{егер } -4<x<4 \\ x^2/5 & \text{егер } 4<x<10 \\ x & \text{егер } 10<x \end{cases}$$

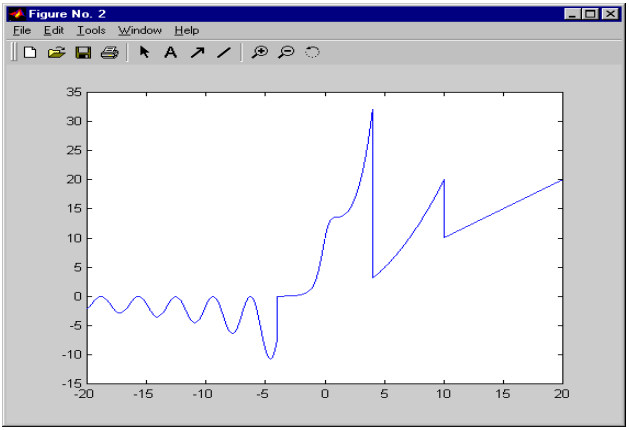
функциясының графигін салу үшін функция құру керек (мәндері dx адыммен есептелінеді). Функция мәтіні:

```
function v=Vg(xn,dx,xk)
x=xn:dx:xk;
y=(x<=-4).*(50*sin(x).^2./x)+(x>-4&x<=4).
*(10*exp(x)./(x.^2+1))
  +(x>4&x<=10).*(x.^2/5)+(x>10).*x;
plot(x,y)
```

Функцияны шақыру:

```
»Vg(-20,0.0001,20)
```

Нәтижесі:



2-мысал. Алдыңғы мысалдағы $y(x)$ функциясы мен $f(x) = x^3 - 7x + 6$ функциясының аралық шектерінің қиылысу нүктелерінің санын анықтау керек. Табылған аралықтардың орталықтарынан шеңбер салу керек. Кометаны пайдаланып, берілген функция және табылған қиылысу нүктелері графиктерімен шектелген жазықтық бөлігін жүргізу. Комета қозғалысын екінші қиылысу нүктесінде бастап-аяқтау керек: бірінші аймақты сағат тілімен бағыттас жүргізу керек, ал екіншісін - сағат тіліне қарсы бағытта.

Функция:

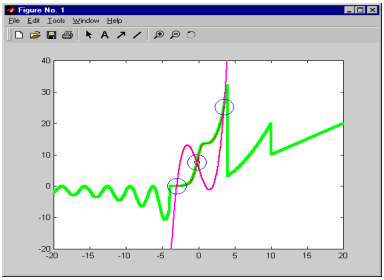
```
function l4_p2_3(xn,xk,dx)
% 2-мысал
% есептеп, график құру
hold off
x=xn:dx:xk;
y1=f1(x);
y2=f2(x);
plot(x,y1,'g','LineWidth',3)
hold on
plot(x,y2,'m','LineWidth',2)
axis([-20 20 -20 40]) % масштабты өзгерту
% графиктегі қиылысу нүктелердің координаттарын
шамамен анықтау
y1_y2=y1-y2;
r=[x(1:end-1); y1_y2(1:end-1).*y1_y2(2:end)<0];
rt=r';
rs=flipud(sortrows(rt,2));
nt=sum(rs(:,2))
xn=rs(1:nt,1)
xk=xn+dx
xr=xn+dx/2;
xr=sort(xr);
yr=f1(xr);
% графиктердегі шамамен табылған қиылысу
нүктелерінің орталығынан шеңбер сызу
scatter(xr,yr,dx*10000)
dx=dx/2; % адымды кеміту
% комета траекториясының координаттарын беру
xc1= xr(2):-dx:xr(1);
xc2=xr(1):dx:xr(2);
xc3=xr(2):dx:xr(3);
xc4=xr(3):-dx:xr(2);
xc=[xc1 xc2 xc3 xc4];
yc=[min([f1(xc1); f2(xc1)]) max([f1(xc2); f2(xc2)])
min([f1(xc3); f2(xc3)]) max([f1(xc4); f2(xc4)]) ];
```

```
comet(xc,yc) % кометаның траектория бойынша
қозғалысы
function y=f1(x)
y=(x<=-4).*(50*sin(x).^2./x)+...
  (x>-4&x<=4).*(10*exp(x)./(x.^2+1))+...
  (x>4&x<=10).*(x.^2/5)+...
  (x>10).*x;
function y=f2(x)
y=x.^3-7*x+6;
```

Нәтижесі (нөлге бөлу туралы ескертусіз):

```
» l4_p2_3(-20,20,0.05)
nt =
     3
xn =
    3.5000
   -0.2500
   -3.0000
xk =
    3.5500
   -0.2000
   -2.9500
```

Алынған график:



3-мысал. Абсциссасы *x1*-ден *x2*-ге дейін және ординатасы *y 1*-ден *y2*-ге дейін болатын координат жазықтығында кездейсоқ түрде n нүктелер беріледі. Сол жақ жоғарғы бұрышының координаттары (*xp1, yp1*), ал оң жақ төменгі бұрышының координаттары (*xp2, yp2*) болатын тік төртбұрышқа (оның шекараларын

қоса) жататын нүктелер санын анықтау керек. Тік төртбұрышты және оған жататын нүктелерді бір түспен боялған дөңгелекпен, ал жатпайтындарын басқа түспен боялған кіші радиустағы дөңгелекпен салу керек. Функция:

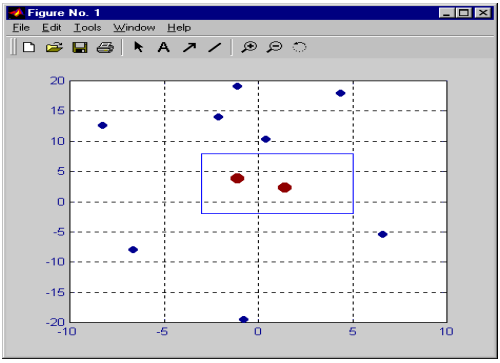
```
function nt=l4_p3(x1,x2,y1,y2,n,xp1,yp1, xp2,yp2)
hold off
xp=[xp1,xp2,xp2,xp1 xp1];
yp=[yp1, yp1,yp2,yp2, yp1];
plot(xp,yp)% тік төртбұрыш салу
hold on
axis([x1,x2,y1,y2])
grid on
x=fv(x1,x2,n);
y=fv(y1,y2,n);
r=x>=xp1&x<=xp2&y>=yp2&y<=yp1;% тік төртбұрышқа
нүктенің жатуы
xy=[x y r]
nt=sum(xy(:,3)); % тік төртбұрышта жататын нүктелер
саны
scatter(xy(:,1),xy(:,2),50+50*xy(:,3),4+xy(:,3)*9,'filled') %
нүктелерді салу
function k=fv(x1,x2,n)
k=x1+(x2-x1)*rand(n,1);
```

Нәтижесі:

```
» l4_p3(-10,10,-20,20,10,-3,8, 5,-2)
xy =
   6.5773   -5.3480         0
  -6.6746   -7.8986         0
  -2.1219   14.0738         0
   0.4151   10.3792         0
   4.3625   17.9904         0
   1.3838    2.3175    1.0000
  -0.7839  -19.4307         0
  -1.0939    3.8471    1.0000
  -8.2451   12.6482         0
  -1.1303   19.0837         0
```

```
ans =
    2
»
```

Графигі:



Бақылау жұмысы

MATLAB-та графиктер салу.

Жұмыстың мақсаты:

1. Функция графигін салу.

2. Диалогтық режимде жұмыс істеудің тәжірибелік жолдарын игеру.

Жұмыстың тапсырмасы

1 есеп.

1) Мәтінді комментарий ретінде және программа тақырыбы ретінде енгізу.

2) Бастапқы мәліметтерді енгізу.

3) Аргументтің өзгерісін беру.

4) 1 және 2 функциялардың берілген аралықтағы мәнін есептеу.

5) Декарттық координаттар жүйесіндегі бір графикке функцияның бірнеше графигін салу. Әр графикті әр түрлі сызықтармен беру.

2 есеп.

1. 1-ші есептің 1...4 пункттері.

2. Бір графиктің екі терезешесіне графиктер шығару. Бағанды форматтағы графиктер.

Тапсырмалар варианты:

№	1 функция	2 функция	a	b	h
1	$y = \sin(x)$	$z = \exp(x+3)/5000 - 1$	-2π	2π	$\pi/20$
2	$y = \cos(x)$	$z = 0.00025e^3\text{-}x - 0.6$	-2π	2π	$\pi/20$
3	$y = \lvert tg(x)\rvert + 0.1$	$z = (1+x)^6$	-2π	2π	$\pi/20$
4	$y = (x^2\text{-}1)/15$	$z = 1+\sin(x)$	-2π	2π	$\pi/20$
5	$y = (x^3\text{-}2)/15$	$z = 5\cos(x)$	-2π	2π	$\pi/20$
6	$y = x^2 - 10$	$z = 0.025\exp(-1.2x)$	-5	5	1
7	$y = 3\sin(x)$	$z=0.015x^3$	-5	5	1
8	$y = 4\sin(x)$	$z = 0.05x^2$	1	10	1
9	$y = 6\sin(x)$	$z = 0.01x^3$	-10	10	1
10	$y = 2+\cos(x)$	$z = - 0.05(x^2 + 10\cos(x))$	-8	8	1
11	$y = \sin^2(x/3)$	$z = 0.01(x^2 - 40\sin(x))$	-8	8	1
12	$y = \cos^3(x)$	$z = \sin(x) + \sin(2x)$	$-\pi$	π	$\pi/8$
13	$y = 0.5x + \cos^2(x)$	$z = \sin^2(x) + \cos(x)$	$-\pi$	π	$\pi/8$
14	$y = \sin(x) + \cos^2(2x)$	$z = x(0.5 + x)\exp(0.1x)$	$-\pi$	π	$\pi/8$
15	$y = \lvert\sin(x)\rvert\exp(x/2)$	$z = 5x - x^{1.5}+\sin(x)$	0	5	0.5

Орындау мысалдары.

Тапсырма:

1-ші функция $y = 2\sin(x)$

2-ші функция $z = 0.02x^3$

Аргументтің бастапқы мәні $a = -2\pi$

Аргументтің соңғы мәні $b = 2\pi$

Аргументтің өзгеру адымы $h = \pi/20$

1 есеп

```
% 1 есеп
% Аралық және адым
a=-2*pi;
b=2*pi;
h=pi/20;
```

```
% Аргументті беру
X=a:h:b;
% Функцияны есептеу
Y=2*sin(X);
Z=0.02*X.^3;
% 1-ші терезеге бірдей адымдағы сызықпен график шығару
figure(1);
plot(X,Y,X,Z);
% Координаталық торды қосу
grid on
% 2-ші терезеге әр типтегі графиктер шығару
figure(2);
plot(X,Y,'-',X,Z,':');
% Координаталық торды қосу
grid on
2 есеп
% 2 есеп
% Аралық және адым
a=-2*pi;
b=2*pi;
h=pi/20;
% Аргументті беру
X=a:h:b;
%Функцияны есептеу
Y=2*sin(X);
Z=0.02*X.^3;
% 1-ші шағын терезеге баған түрінде 1-ші графикті шығару
subplot(2,1,1),bar(X,Y);
% 2-ші шағын терезеге баған түрінде 2-ші графикті шығару
subplot(2,1,2),bar(X,Z);
```

5. БАҒДАРЛАМАЛАУ

5.1. Функцияны енгізу және шығару

Disp(айнымалы атауы) функцияның атауын көрсетпей-ақ айнымалы мәнін экран бетінен енгізуге арналған. Disp функциясының аргументі дәйекшелерге алынған мәтіндік жолда болуы мүмкін: disp(‘символдық жол‘). Әрбір disp функциясын шақыру жаңа жолға ауысуды іске асырады.

Input функциясы экранға сұраныс жасау үшін және бір айнымалының мәнін енгізуді қорытындылау үшін арналған немесе MATLAB жүйесіндегі (енгізілетін функция өрнектері шақырылған функция мәні болуы мүмкін) өрнектерді кері қайтарады. Функцияны шақыру келесідей түрде жазылады: айнымалы атауы =input(‘шақыру’). Қайтарылатын функцияның айнымалы мәнін тағайындайды, оның атауы шақыру арқылы берілген (жиымды енгізу кезінде оның элементтері тік жақшада және матрицалар жолдардың соңында енгізу керек;). Мысалы, inp.m: файлда сақталған мәтін сценарийі:

```
x=input('x айнымалы мәнін енгізіңіз')
s=input('s айнымалы мәнін енгізіңіз')
v=input('v  векторын енгізіңіз ')
h=input('h матрицаны енгізіңіз ')
```

Нәтижесі:

```
» inp
айнымалы мәнін енгізіңіз x 4
x =
    4
айнымалы мәнін енгізіңіз s sin(0.3)/2+x^2-7
s =
   9.1478
v [3 5 1] векторды енгізіңіз
v =
    3    5    1
h [2 4; 7 8] матрицаны енгізіңіз
h =
```

```
    2    4
    7    8
»
```

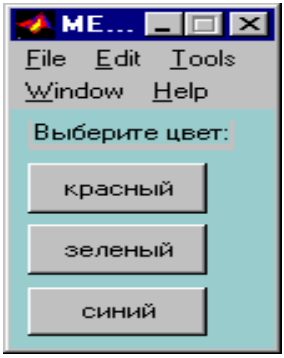
Input функциясын шақыру келесідей түрде жазылады: айнымалы атауы=input(‘шақыру’,‘s’). Мысалы:

```
» h=input('Өзіңіздің атыңызды енгізіңіз','s')
Өзіңіздің атыңызды енгізіңіз ГҮЛНАР
h =
ГҮЛНАР
»
```

Menu функциясын шақыру келесі түрде жазылады: айнымалы атауы=menu(‘мәзір тақырыбы’,‘1 таңдау’,‘2 таңдау’, … ,’n таңдау’). Экран бетіне таңдау батырмалары, мәзір және тақырып шығарылады. Функция таңдалған батырма нөмірін қайтарады (функцияны шақыру кезіндегі тізімде). Мысалы:

```
» k=menu('Түсті таңдаңыз:','қызыл','жасыл','көк')
k =
    3
»
```

Функцияларды орындаған кезде экранға келесідей мәзір пайда болады:



«Көк» батырмасын басқанда функция 3 мәнін қайтарды.

5.2. Циклдік операторлар

MATLAB-та екі циклдік оператор бар: for операторы берілген санның циклдік орындалуын қамтамасыз етеді, while циклдік операторлар тобын логикалық өрнектің мәніне сәйкес орындайды. Циклдік операторлар есептеулердің командалық терезесінде, сценарийлер мен функцияларда қолданылады.

5.2.1. For циклдік операторы

Циклдік операторлардың жазылуы:

for цикл параметрі=бастапқы мән:адым:соңғы мән
(циклдік дене) операторлар
end

Цикл денесі цикл параметрінің бастапқы мәннен соңғы мәніне дейінгі мәндерді қабылдау аралығында жұмыс істейді. Егер адым бірге тең болса, оны көрсетпеуге де болады. Циклдік дене операторлары әртүрлі жолдарда немесе бір жолда орналаса алады (бұл жағдайда оларды үтір немесе нүктелі үтір арқылы ажыратуға болады). Мысалы, берілген адым мен аралықтағы сандардың туындысы мен қосындысын анықтау үшін функция құрастырамыз:

```
function [s,p]=sp(in,ik,di)
s=0;
p=1;
for i=in:di:ik
  s=s+i;
  p=p*i;
end
```

Нәтижесі:

```
» [S,P]=sp(3,15,2)
S =
   63
P =
    2027025
»
```

Цикл параметріне мән беру үшін вектор-жолды пайдалануға болады:

```
for цикл параметрі=А жиымы
(циклдік дене) операторлар
end
```

Егер А вектор-жол болса, онда цикл параметрі вектор элементтерінің мәнін дәйекті қабылдайды. Мысал ретінде вектор элементтерінің мәндерінің қосындысын анықтайтын функция құраймыз.

```
function s=sv(v)
s=0;
for i=v
  s=s+i;
end
```

Нәтижесі:

```
» sv([4 6 3])
ans =
   13
»
```

Егер *A* матрицасының әрбір қатары цикл параметрі болса, онда матрицада қанша қатар болса, сонша цикл орындалады.

Алдында құралған вектор элементтерінің қосындыларының мәнін есептеу функциясын қолданып, матрица қосындысының мәнін анықтайтын функция құрастырайық:

```
function s=sm(a)
s=0;
for i=a
  s=s+sv(i');% бағананы тасымалдаймыз
end
```

Нәтижесі:

```
» f=[3 4; 5 7; 2 1]
f =
    3    4
    5    7
    2    1
» sm(f)
ans=
   22
```

5.2.2. While циклдік операторы

While циклдік операторының жазылуы келесі түрдегідей болады.

while логикалық өрнек операторы
(циклдік дене) операторлар
end

Цикл денесі логикалық өрнек шынайы болғанда орындала береді. Логикалық өрнектің мәні жиым болса және оның барлық элементтері шынайы болса (нөлге тең емес), сонымен қатар жиым бос болмаса, ([] - бос жиым), онда оның мәні шынайы болады. Мысалы: Берілген ε=0,01, i=0,00000005 болғанда,
ε (ε>0) дәлдікте

$$\sum_{i=1}^{\infty} \frac{1}{i(i+1)}$$

өрнегінің функциясын құру керек. Функция мәтіні (функцияның қосындысынан басқа итерациялардың санын қайтарады):

```
function [s,i]=sr1(eps)
s=0;
i=0;
y=1;
while abs(y)>eps
  i=i+1;
  y=1/i/(i+1);
  s=s+y;
end
```

Нәтижесі:

```
» [s1,n1]=sr1(0.01)
s1 =
   0.9091
n1 =
   10
» [s2,n2]=sr1(0.0000000005)
```

```
s2 =
   1.0000
n2 =
     44721
```

Егер қайтарылатын мәндерде вектор қалыптаспаған болса, онда функция тек қана біріншісін қайтарады:

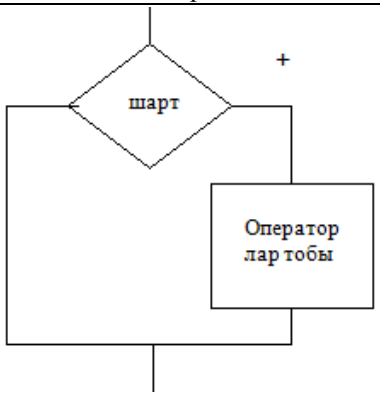
```
» sr1(0.0000000005)
ans =
   1.0000
```

5.3. Тармақталған оператор

Тармақталған операторға ауыстырып-қосу операторы және шартты оператор жатады.

5.3.1. Шартты оператор

Шартты оператор келесі формалардың біреуіндегі түрде жазылуы мүмкін:

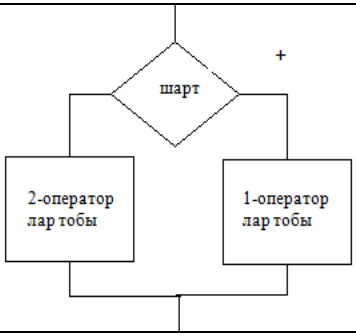
Синтаксис	Блок-сұлбасы
if шарт операторлар тобы end	

```
if шарт

1 операторлар
тобы

else
2 операторлар
тобы

end
```




```
if 1 шарт

1 операторлар
тобы

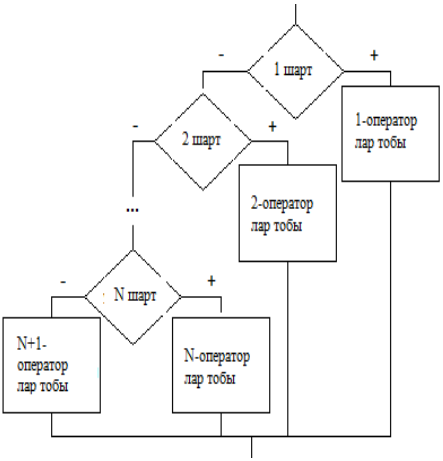
elseif 2 шарт

2 операторлар
тобы
…
elseif N шарт

N операторлар
тобы

else
N+1 операторлар
тобы

end
```



Шарт ретінде кез келген өрнек қолданылуы мүмкін (әдетте, логикалық өрнектер қолданылады). Егер шарт ретінде жиым қолданылса, ол жиымның барлық элементтері нөлге тең және бос болмаса, онда ол «ақиқат» болады. Мысалы, үш саннан көп санды анықтау үшін max функциясын құраймыз:

```
function max=l5_p1(a,b,c)
if a>b
  if a>c
    max=a;
  else
    max=c;
  end
elseif b>c
  max=b;
else
  max=c;
end
```

Нәтижесі:

```
» l5_p1(13, 4, 5)
ans =
   13
```

5.3.2. Ауыстырып-қосқыш оператор

Ауыстырып-қосқыш операторы келесі түрде жазылады.

```
switch В1 айнымалы немесе өрнек
case 1-ші мағынасы
1 операторлар тобы
case { 2-ші мағынасы, 3-ші мағынасы, 4-ші мағынасы, ... }
2 операторлар тобы
...
otherwise
 n операторлар тобы
end
Z оператор
```

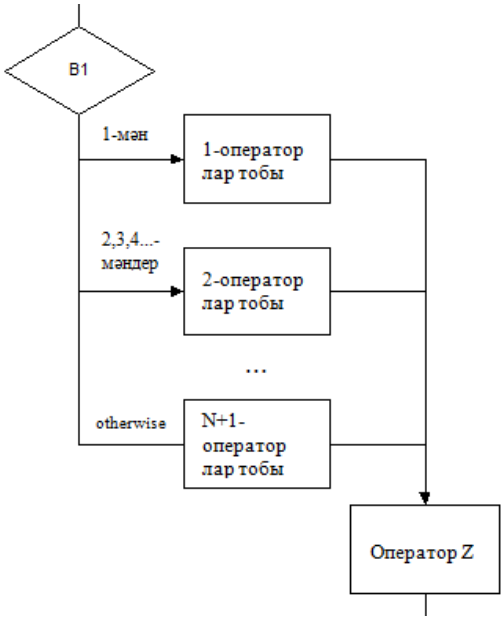
Switch түйінді сөзі жазылғаннан кейін, өрнек пен айнымалы мәніне тәуелді болуын тармақталған оператор ұйымдастырады. Case тобының саны әрқалай бола береді. Case түйінді сөзінен кейін өрнектің мәні немесе фигуралы жақшаға алынған бірнеше мәндер жазылады.

Оператор келесідей жағдайларда жұмыс істейді:

- switch түйінді сөзінен жазылғаннан кейінгі өрнектің мәнін есептейді;

- егер өрнек мәні case түйінді сөздерінің бірінен кейінгі жазылған мәнге сәйкес келсе, онда сол case түйінді сөзінен кейінгі операторлар тобы кезекпен орындалады. Әрі қарай end түйінді сөзінен кейінгі операторларға өту орындалады. Егер case түйінді сөзінен кейін бірнеше мән фигуралы жақша ішінде жазылса, онда операторлар тобы орындалады;

- егер өрнек мәндері ешбір case түйінді сөзінен кейінгі жазылған мәнімен сәйкес келмесе, онда otherwise түйінді сөзінен кейінгі операторлар тобы орындалады (егер ол операторлар бар болса). Блок-сұлбасы:



Мысалы, $\sin(x), x^2, e^x$ функцияларының біреуінен график құру функциясын құраймыз:

```
function graph3
v=menu('Функцияны енгізіңіз','sin(x)','x^2', 'exp(x)');
xn=input('аргументтің бастапқы мәнін енгізіңіз');
xk=input('аргументтің соңғы мәнін енгізіңіз' );
dx=(xk-xn)/500;
x=xn:dx:xk;
switch v
case 1
  y=sin(x);plot(x,y)
case 2
  y=x.^2;plot(x,y)
case 3
  y=exp(x);plot(x,y)
otherwise %егер мәзірді шақырғаннан кейін v айнымалының
мәні өзгереді.
  disp('дұрыс емес таңдау')
end
```

Дәл сол мысал жолдық айнымалы үшін:

```
function graph4
v=input('функцияны енгізіңіз','s');
xn=input('аргументтің бастапқы мәнін енгізіңіз');
xk=input('аргументтің соңғы мәнін енгізіңіз');
dx=(xk-xn)/500;
x=xn:dx:xk;
switch v
case 'sin(x)'
  y=sin(x);;
  plot(x,y)
case 'x^2'
  y=x.^2;
  plot(x,y)
case 'exp(x)'
  y=exp(x);
  plot(x,y)
otherwise
 disp('дұрыс емес таңдау')
end
```

Мәзірдің таңдаулы тармағы және мәзірге қайтаруына байланысты одан шығатын тармақтың орындалуын ұйымдастыру мысалы:

```
function l5_p2
v=1
while v~=3
  v=menu('таңдау','1','2','шығу')
  switch v
  case 1
    disp (1)
  case 2
    disp(2)
  case 3
   disp( 'ШЫҒУ')
  end
end
disp('СОҢЫ')
```

5.4. Басқа операторлар

Break операторы for және while циклдерін тоқтады. Мысалы:

```
for i=1:1:5
  for j=1:5
    if i==2& j==3
      break
    end
  end
   if i==2
   break
  end
end
i
j
```

Return операторы шақырылған функцияны кері қайтару немесе пернемен жұмыс істеу режимін орындайды.

```
function [i,j]=return1
for i=1:1:5
  for j=1:5
    if i==2& j==3
      i
      j
      return
    end
  end
end
```

Нәтижесі:

```
» [z,x]=return1
i =
    2
j =
    3
z =
    2
x =
    3
»
```

Бағдарламаны орындау кезінде кез келген пернені басқанға дейін үзіліс жасау үшін pause командасы қолданылады. Pause(n) командасы n секундта жұмыс жүйесін тоқтатады. Pause on командасы үзілісті іске қосса, pause off командасы оның жұмысын тоқтатады. Мысалы:

```
function pause1
xn=input('аргументтің бастапқы мәнін енгізіңіз');
xk=input('аргументтің соңғы мәнін енгізіңіз');
dx=(xk-xn)/500;
x=xn:dx:xk;
for v=1:3
 subplot(2,2,v)
  switch v
  case 1
```

```
    y=sin(x);
    plot(x,y)
  case 2
    y=x.^2;
    plot(x,y)
  case 3
    y=exp(x);
    plot(x,y)
  end
  pause
end
```

5.5. Мысалдар

1-мысал. Жазықтықтарда n нүкте берілген. Ең кіші массасы бар нүктенің кейбіреуі өзіне ең жақын тұрған нүктеге өз массасын алып беріп, өзі жойылады. Ол үдеріс жалғыз нүкте қалғанша жүреді. Сол нүктенің координатасын анықтау керек. Функциялардың мәтіндері:

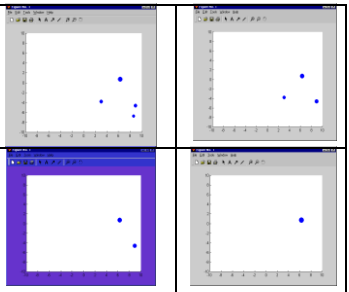
```
function l5_p3_s_1(n,x1,x2,y1,y2,m1,m2)
x=fv(x1,x2,n);
y=fv(y1,y2,n);
m=fv(m1,m2,n);
[X,Y]=l5_p3_1(x,y,m,x1,x2,y1,y2)
function k=fv(x1,x2,n)
k=x1+(x2-x1)*rand(n,1);
```

```
function [X,Y]=l5_p3_1(x,y,m,xn,xk,yn,yk)
a=[x y m];
s=size(a);
scatter(x,y,sqrt(m/pi)*100,'filled')
axis([xn,xk,yn,yk])
for i=1:s(1)-1
  [min,imin]=min(a(:,3));
  b=a(imin,:);
pause
  a(imin,:)=[];
```

```
    a(:,4)=sqrt( (a(:,1)-b(1)).^2+ (a(:,2)-b(2)).^2 );
    [rmin,ib]=min(a(:,4));
    a(ib,3)=a(ib,3)+b(3)
    scatter(a(:,1),a(:,2),sqrt(a(:,3)/pi)*100,'filled')
    axis([xn,xk,yn,yk])
  end
  X=a(1);
  Y=a(2);
```

Нәтижесі:

```
» l5_p3_s_1 (4,-10,10,-10,10,1,10)
a =
   3.0310   -3.8012    2.8989    6.2986
   8.9217   -4.6237    4.4263    2.1344
   6.3177    0.7290    6.8661    7.8061
a =
   8.9217   -4.6237    4.4263    5.9479
   6.3177    0.7290    9.7650    5.5969
a =
   6.3177    0.7290   14.1913    5.9525
X =
   6.3177
Y =
   0.7290
»
```



2-мысал. Бастапқы матрицаның нөлдік элементтерінің индекстерін және санын анықтау керек. Функция (нөлдік элементтердің саны - *k*, *a* жиымында ол нөлдік элементтердің индексі, бірінші бағанда – жолдың нөмірі, екінші бағанда - бағанның нөмірі):

```
  function [a,k]=l5_p4(b)
  s=size(b);
  k=0;
  for i=1:s(1)
    for j=1:s(2)
      if b(i,j)==0
```

```
        k=k+1;
        a(k,1)=i;
        a(k,2)=j;
      end
    end
end
```

Нәтижесі:

```
» h=[4 0 -3; 6 8 0; 0 0 0],[R,K]=l5_p4(h)
h =
    4    0   -3
    6    8    0
    0    0    0
R =
    1    2
    2    3
    3    1
    3    2
    3    3
K =
    5
»
```

Бақылау жұмысы.

MATLAB-та көпөлшемді есептеулер.

Жұмыстың мақсаты:

1. MATLAB пакетінің негізгі жағдайларымен танысу.
2. Циклдарды ұйымдастыру тәртіптерін игеру.
3. Көпөлшемді нәтижелер алу тәртіптерін игеру.
4. Көпөлшемді мәліметтерді кесте түрінде шығару әдістерін игеру.
5. Диалогтық режимде жұмыс істеудің тәжірибелік жолдарын игеру.

Жұмыстың тапсырмасы

1 есеп. Екіөлшемді функция және көлемдік график салу.

- Бастапқы мәліметтерді енгізу.
- Екіөлшемді функцияны есептеу.

- Функцияның әр түрлі типтегі 5 үшөлшемді графигін шығару.
- Функцияның әр түрлі типтегі 2 контурлық графигін шығару.

2 есеп. Екіөлшемді функция және жалпы терезенің шағын терезелеріне көлемдік графиктер салу.

Тапсырмалар варианты

№	Функция	Өзгеру шектері	
		x	y
1	z=sin(x)cos(y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
2	z=sin(x/2)cos(y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
3	z=sin(2x)cos(y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
4	z = sin(x)cos(y/2)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
5	z = sin(x/2)cos(2y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
6	z = sin(2x)cos(2y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
7	z = 1+sin(x)/x)(sin(y)/y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
8	z = (sin(x)/x)cos(y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
9	z = (sin(x)/x)|cos(y)|	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
10	z = (sin(x)/x)y	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
11	z = (sin(x)/x)|y|	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
12	z = (sin(x)/x)sin(y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
13	z = (sin(x)/x)|sin(y)|	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
14	z = (sin(x)/x)(1-y)	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін
15	z = (sin(x)/x)|y+0.5|	-2π-ден 2π-ге дейін	-2π-ден 2π-ге дейін

Орындау мысалы.

Тапсырма.

$z = \frac{\sin(x)}{x} \cdot \frac{\sin(y)}{y}$ функциясы берілген. Аргументтерінің өзгеру аралығы -2π...2π.

1 есеп

```
% 1 есеп
% Нүкте мен адымдар саны
N=40;
h=pi/20;
% Матрица есептеу
for n=1:2*N+1
 if n==N+1 A(n)=1; else A(n)=sin(h*(n-N-1))/(h*(n-N-1)); end;
end;
for n=1:2*N+1
   for m=1:2*N+1
      Z(n,m)=A(n)*A(m);
   end;
end;
% Аймақты беру
[X,Y]=meshgrid([-N:1:N]);
% 1-ші терезеге аксонометрияда график шығару
figure(1);
plot3(X,Y,Z);
% 2-ші терезеге функционалды бояуымен үшөлшемді график шығару
figure(2);
mesh(X,Y,Z);
% 3-ші терезеге функционалды бояуымен үшөлшемді график шығару
figure(3);
meshc(X,Y,Z);
% 4-ші терезеге проекциямен үшөлшемді график шығару
figure(4);
surf(X,Y,Z);
% 5-ші терезеге контурлық график шығару
```

```
figure(5);
contour(X,Y,Z)
% 6-шы терезеге көлемдік контурлық график шығару
figure(6);
contour3(X,Y,Z)
% 7-ші терезеге жарығымен көлемдік график шығару
figure(7);surfl(X,Y,Z)
2-есеп
% 2 есеп
% Нүкте мен адымдар саны
N=40;
h=pi/20;
% Матрица есептеу
for n=1:2*N+1
 if n==N+1 A(n)=1; else A(n)=sin(h*(n-N-1))/(h*(n-N-1)); end;
end;
for n=1:2*N+1
   for m=1:2*N+1
      Z(n,m)=A(n)*A(m);
   end;
end;
% Аймақты беру
[X,Y]=meshgrid([-N:1:N]);
% 1-ші шағын терезеге аксонометрияда график шығару
subplot(3,3,1),plot3(X,Y,Z);
% 2-ші шағын терезеге функционалды бояуымен үшөлшемді график шығару
subplot(3,3,2),mesh(X,Y,Z);
% 3-ші шағын терезеге функционалды бояуымен және проекциямен үшөлшемді график шығару
subplot(3,3,3),meshc(X,Y,Z);
% 4-ші шағын терезеге проекциямен үшөлшемді график шығару
subplot(3,3,4),surf(X,Y,Z);
% 5-ші шағын терезеге контурлық график шығару
subplot(3,3,5),contour(X,Y,Z)
% 6-шы шағын терезеге көлемдік контурлық график шығару
```

```
subplot(3,3,6),contour3(X,Y,Z)
% 7-ші шағын терезеге жарығымен көлемдік график шығару
subplot(3,3,7),surfl(X,Y,Z)
```

Бақылау сұрақтары

1. Екіөлшемді графиктерді жасау тәртібін түсіндіріңіз.
2. Бір терезеде бірнеше графикті қалай салады?
3. Үшөлшемді графикті функционалды бояуымен және проекциясымен қалай салуға болады?
4. Контурлық графика деген не?
5. Графикалық терезеге жазулар, атаулар, координаталық торлар қалай шығарылады?
6. Графиктің масштабын қалай өзгертуге болады?

6. MATLAB-ТАҒЫ ТИПТІК ЕСЕПТЕР

6.1. Сызықтық емес теңдеулерді және теңдеулер жүйесін шешу $f(x)=0$ теңдеуінің түбірін анықтау үшін fzero(fun, x0, tol) түрдегі функциясы қолданылады. Мұндағы, fun – түбірін табу керек функция, x0 - түбірінің бастапқы жуықтауы. fzero функциясы түбірдің жуық мәнін қайтарады; tol – анықталатын түбірдің қателігі (егер қателік берілмесе, ол eps-ге тең болып саналады). fun функциясы MATLAB ережесі бойынша формула ретінде беріледі, дәйекшеде немесе М-функция атымен теңдеудің сол жағын анықтауыш ретінде жазылады. Бастапқы жуықтау скалярлық түрде немесе аралық түрінде берілуі мүмкін. Функция соңында әртүрлі белгі арқылы беріледі (векторлық координат басы және аралық соңы). Мысалы, $2^x+\sin(x)+x=0$ теңдеуінің түбірін анықтайық. Теңдеудің сол жағын есептеу үшін f7 аламыз:

```
function y=f7(x)
y=2.^x+sin(x)+x;
```

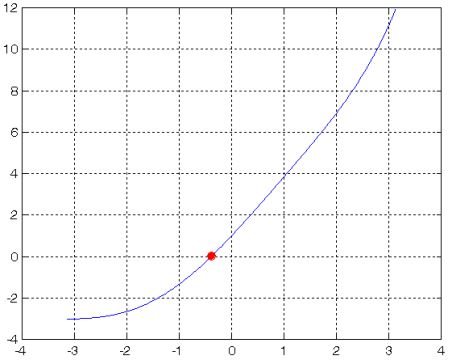
Келесі fzero функциясын әртүрлі мәнде аламыз:

```
x=-pi:0.01:pi;
plot(x,f7(x));
grid on
fzero('2.^x+sin(x)+x',-0.5)
fzero('2.^x+sin(x)+x',-0.5,1e-3)
fzero('f7',-0.5,1e-4)
xr=fzero('f7',[-1,0],1e-4);
hold on;
scatter(xr,f7(xr),40,'r','filled');
```

Нәтижесі:

```
» l7_p0
Zero found in the interval: [-0.38686, -0.58].
ans =
  -0.3871
ans =
  -0.3869
ans =
  -0.3871
»
```

Графигі:

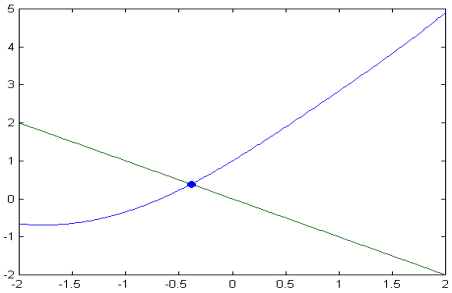


fzero функциясы берілген функцияның абсцисса осіндегі қиылысқан нүктелерін анықтайды. Функцияның абсцисса осімен қиылысқан нүктелері 0 болып есептелмейді.

Теңдеулер мен теңдеулер жүйесін символдық түрде шешу үшін solve функциясы қолданылады. Ол функцияны қолданғанда, теңдеу доғал жақшаға алынып, дәйекшемен үтір арқылы жазылады. Мысалы, $2^x + \sin(x) = -x$ теңдеуінің графикалық шешімін мына түрде орындауға болады:

```
» xr=solve('2^x+sin(x)=-x'); x=-2:0.01:2;y1=2.^x+sin(x);
y2=-x;...
plot(x,y1,x,y2);hold on; scatter(double(xr),double
(-xr),40,'filled')
» xr
xr =
-.38712465392757487234848391052801
»
```

Алдағы суретте $y = 2^x + \sin(x)$ және $y = -x$ функцияларының графигі және олардың қиылысқан нүктелері көрсетілген:



Келтірілген мысалда *xr* символдық айнымалы қолданылады, сол себепті нүктенің құрылуында double типі қолданылады (double (xr)).

Келесі мысал теңдеулер жүйесінің шешімін көрсетеді:

$$\begin{cases} \sin(x+y)-1.2x=0.2 \\ x^2+y^2=1 \end{cases}$$

```
» [xr,yr]=solve('sin(x+y)-1.2*x=0.2','x^2+y^2=1')
xr =
-.95682502589007371450373183654668
yr =
-.29066453142834570132077207025201
»
```

Тексеру:

```
» x=double(xr)
x =
    -0.95683
» y=double(yr)
y =
    -0.29066
» abs(sin(x+y)-1.2*x-0.2)<eps
ans =
    1
» abs(x^2+y^2-1)<eps
ans =
    1
```

6.2. Бір немесе бірнеше айнымалысы бар функциялардың минимумын анықтау

Функцияның минимумын табу бір айнымалысы бар fmin функциясының көмегімен іске асырылады және келесі түрде жазылады:

xmin=fmin(‘функция атауы’, x1,x2)

xmin=fmin(‘функция атауы’, x1,x2, options)

xmin=fmin(‘функция атауы’, x1,x2, options, p1, p2, …, p10),

мұндағы, функция атауы – табу керек минимумның М-функциясының атауы; x1, x2- аралықтың басы және соңы; options – басқаратын параметр векторы, ол оптимизациялау алгоритмнің құрылғысына арналған, егер бәрі қалыпты күйде қалса, онда бұл аргумент орнына бос жиым жазылады.

p1, p2,…, p10 – функция параметрлері. Мысалы, функцияның минимумдарын аралық арқылы анықтау:

Функция	Аралық
cos(x)	[1 , 4]
cos(p1*x+p2) егер p1=2, p2=1	[1 , 4]
cos(p1*x+p2) егер p1=1, p2=2	[5 , 8]

М-функциясы:

```
function y=l7_p9f1(x,p1,p2)
y=cos(x*p1+p2);
```

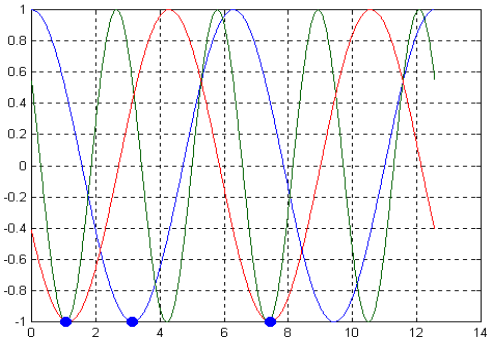
Сценарийдің мәтіні:

```
xmin1=fmin('cos',1,4)
xmin1_2=fmin('l7_p9f1',1,4,[],1,0)
xmin2=fmin('l7_p9f1',1,4,[],2,1)
xmin3=fmin('l7_p9f1',5,8,[],1,2)
x=0:0.01:pi*4;
plot(x,[cos(x);l7_p9f1(x,2,1);l7_p9f1(x,1,2)])
hold on
grid on
scat ter([xmin1;xmin2;xmin3],[cos(xmin1);l7_p9f1(xmin2,2,1);
l7_p9f1 (xmin3,1,2)],70,'filled')
```

Нәтижесі:

```
» l7_p9
xmin1 =
        3.14158640896816
xmin1_2 =
        3.14158640896816
xmin2 =
        1.07078246937276
xmin3 =
        7.42479085012525
```

Графигі:



Бірнеше айнымалысы бар функцияларды минимизациялау келесі функциямен анықталады: xmin=fmins('функция атауы',x0).

Айнымалыдағы xmin векторын осы функция қайтарады, функция мұнда бастапқы жуықтаудың минимальді мәндерін *x0* векторында қабылдайды.

$f(x,y)=\cos(x)*\sin(y)$ функциясының минимумының координаттарын (3,-5) нүктелер арқылы арқылы табамыз. График жоғарғы түрде салынып, сызықтары минимум нүктесін көрсетеді. Сценарий:

```
close all;
x=-2*pi:0.3:2*pi;
y=x;
x0=[3 ,-5];
xmin=fmins('l7_p10f2',x0)
[xx,yy]=meshgrid(x,y);
subplot(1,2,1)
mesh(xx,yy,l7_p10f1(xx,yy))
axis([-2*pi,2*pi,-2*pi,2*pi,-1,1])
hold on
scat
ter3(xmin(1),xmin(2),l7_p10f1(xmin(1),xmin(2)),70,2,'filled')
subplot(1,2,2)
contour(xx,yy,l7_p10f1(xx,yy))
grid on
hold on
scat
ter3(xmin(1),xmin(2),l7_p10f1(xmin(1),xmin(2)),70,2,'filled')
```

l7_p10f1 және l7_p10f2 функциясы:

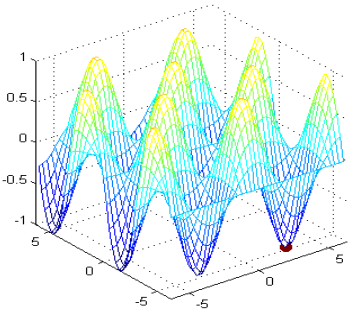
```
function f=l7_p10f1(x,y)
f=cos(x).*sin(y);
```

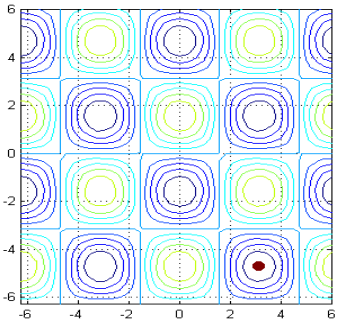
```
function f=l7_p10f2(x)
f=cos(x(1)).*sin(x(2));
```

Нәтиже:

```
» l7_p10
xmin =
        3.14155102269033        -4.71242727650315
```

Алынған график:





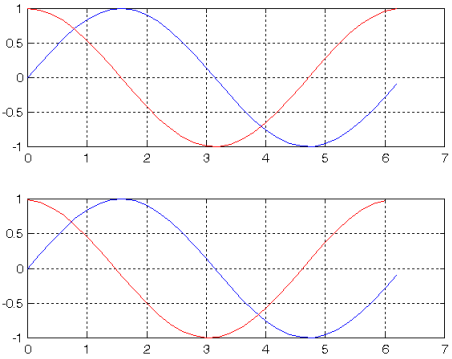
6.3. Сандық дифференциалдау мен шектік айырымды есептеу

diff(x,n) функциясы шектік айырымды есептейді. Егер n элементтен тұратын x аргументі бірөлшемді жиым болса, функция бірөлшемді жиымды [x(2)-x(1) x(3)-x(2)…x(n)-x(n-1)] қайтарады. Қайтарылатын жиымдағы элементтер саны n-1.
diff(x,n) функциясы diff(x,n)=diff(diff(x,n-1)) қатынасындағы n ретті шектік айырымды есептейді.

```
» y=[4 3 7 1]; dy=diff(y)
dy =
   -1    4   -6
» ddy=diff(dy)
ddy =
    5  -10
» ddyy=diff(y,2)
ddyy =
    5  -10
»
```

Кесте бойынша берілген жуық мән алғашқы функцияның туындысы y(x) бойынша diff(y)/diff(x) қатынасына байланысты.

Мысалдағы бірінші терезеде sin(x) функциясының 1-ші реттік туындысының функциясының графигі берілген.



Сценарий:

```
close all
x=0:0.2:2*pi; y= sin(x);
r=size(x);
n=r(2);

dy=cos(x);d2y=-sin(x);
subplot(2,1,1);
plot(x,y);
hold on;
grid on;
plot(x,dy,'r');
plot(x,d2y,'g');

d1=diff(y)./diff(x);  d2=diff(d1)./diff(x(1:n-1));
subplot(2,1,2);
plot(x,y);
hold on;
grid on;
plot(x(1:n-1),d1,'r');
plot(x(1:n-2),d2,'g');
```

6.4. Сандық интегралдау

y функциясынан *x* айнымалысы бойынша алынған интегралдың мәнін трапеция әдісімен есептейтін trapz(x,y) функциясы кері қайтарады. Мысалы, $y = x$ функциясының интегралын 1-ден 2-ге дейінгі аралықта есептейміз:

```
x=1:0.1:2;
y=x;
i=trapz(x,y)% сандық интегралдау
ik=2^2/2-1/2% тексеру
```

Нәтижесі:

```
» l7_p2
i =
      1.5
ik =
      1.5
```

Егер функция бір ғана trapz(y) аргументімен шақырылса, онда 1-ге тең интегралдың тұрақты адымы есептеледі (адымның туынды мәніне ие болу үшін интегралдың алынған мәнін интегралдаудың адым санына көбейту керек), яғни

```
x=1:0.1:2;
y=x;
i=trapz(y)% сандық интегралдау
i=i*0.1
ik=2^2/2-1/2% тексеру
```

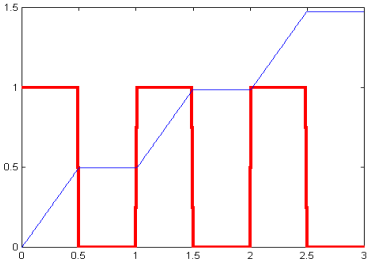
Нәтижесі:

```
» l7_p3
i =
   15
i =
      1.5
ik =
      1.5
```

Ал cumtraps *(x,y)* (cumtraps(y)) функциялары қосымша өтпелі нәтижелердің мәнін есептейді. Тікбұрышты импульс түріндегі (графикте жуан қызыл түспен бейнеленген) функция үшін cumtraps функциясы ізделінген интегралдың жинақталуын көрсететін функцияны көрсетуге көмектеседі. *t1* туынды уақыт моментіндегі мәні 0-ден t1-ге дейінгі аралықтағы интегралдың мәніне тең. Сценарий мәтіні:

```
x=0:0.01:3;
y=(x<0.5|x>1&x<1.5|x>2&x<2.5);
i=cumtrapz(x,y)% сандық интегралдау
plot(x,y,'r','LineWidth',3)
hold on
plot(x,i)
```

Графигі:



quad және quad8 функциясы интегралды квадраттау әдісі бойынша есептеуге көмектеседі (quad функциясы Ньютон-Котес формуласы бойынша екінші реттік болып саналады).

Бір типтік функциялар

quad функциясы	quad8 функциясы
quad('функция атауы', a, b) quad('функция атауы', a, b, tol) quad('функция атауы', a, b, tol, trace) quad('функция атауы', a, b, tol, trace, p1, p2, …)	quad8('функция атауы', a, b) quad8(…)

мұндағы *a, b* – төменгі және жоғарғы интегралдың шектері; *tol* – интегралдаудың салыстырмалы қателігі, тұрақты 1e-3; егер trace ≠ 0 – қосымша нақты интеграласты функциясын орындауға қажет; p1, p2, … - интеграласты функциясының параметрлері.

Бұл функцияларды $\int_{-\pi/2}^{\pi/2} \cos^2(x)dx$ интегралын есептеу үшін қолданайық. Сценарий мәтіні:

```
close all;
i1=quad('l7_p5f',-pi,pi);
i1_8=quad8('l7_p5f',-pi,pi);
i2=quad('l7_p5f',-pi,pi,1e-1);
i2_8=quad8('l7_p5f',-pi,pi,1e-10,1);
disp(['i1=',num2str(i1,20),'
 i1_8=',num2str(i1_8,20),'
  i2='num2str(i2,20),'   i2_8='num2str(i2_8,20)])
```

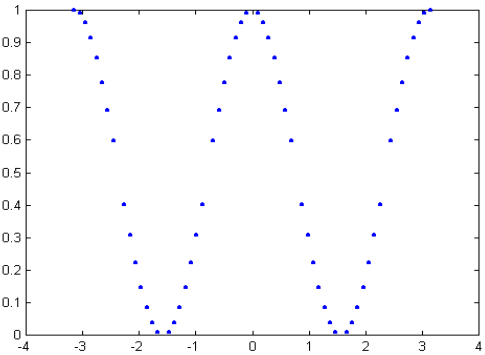
М-функциясы интеграласты өрнегінің мәнін есептеу үшін

```
function y=l7_p5f(x)
y=cos(x).^2;
```

Нәтижесі:

```
» l7_p5
i1=3.1415926535897931   i1_8=3.1415926535897931
i2=3.1415926535897931   i2_8=3.1415926535897931
```

Интеграласты функциясының нүктелік графигі:



$\int_{y1}^{y2}\int_{x1}^{x1} f(x,y)dxdy$ түрдегі екі еселі интералды шешуді dblquad функциясы атқарады.

dblquad:

dblquad('функция атауы', x1, x2, y1, y2)

dblquad('функция атауы', x1, x2, y1, y2, tol)

dblquad('функция атауы', x1, x2, y1, y2, tol, metod)

мұндағы 'функция атауы' – M – функциясының атауы, интеграласты функциясының мәнін береді. Оның екі параметрі болады: біріншісі ішкі интегралдағы айнымалының векторлық мәні, екіншісі сыртқы интегралдағы айнымалының скалярлық мәні. Интегралданатын функция векторды қайтару керек; *x1, x2* – ішкі интегралдың төменгі және жоғарғы шектері; *y1, y2* – сыртқы интегралдың төменгі және жоғарғы шектері;

Мысалы, $\int_{0}^{2\pi}\int_{0}^{\pi} \sin(x)^2 \cdot \cos(x)^2 \cdot y\, dx\, dy$ интегралының мәнін табамыз. M–функциясының мәтіні:

```
function f=l7_p6f(x,y)
f=sin(x).^2.*cos(x).^2*y;
```

Функцияны шақыру және есептеудің нәтижелері:

```
» i=dblquad('l7_p6f',0,pi,0,2*pi)
i =
      7.75156917007495
» i=dblquad('l7_p6f',0,pi,0,2*pi,'quad8')
i =
      7.75156917007495
```

6.5. Көпмүшелерге қатысты есептерді шығару

Полином немесе көпмүше $P(x)=a_1x^n+a_2x^{n-1}+a_3x^{n-2}+...+a_nx+a_{n+1}$ түрінде беріледі.

Көпмүше коэффициенттердің векторы арқылы беріледі (айнымалы дәрежесінің кему ретімен) $A=[a_1,a_2,..a_{n+1}]$ (вектор

элементінің коэффициент саны ретінде көпмүше дәрежесіне бір санға артық болатын 0-дік коэффициенттер есепке алынады).

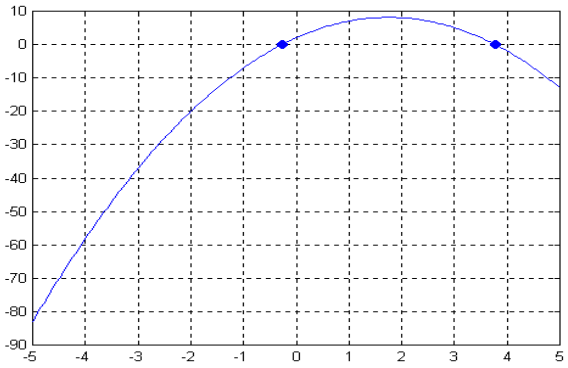
Polyval(A,x) функциясы *x* аргументінде коэффициенті *A* векторымен бірге дәрежелі көпмүшенің мәнін қайтарады:

```
» h=[-2 7 2];polyval(h,3)
ans =
    5
» -2*3^2+7*3+2
ans =
    5
```

Егер *x* - жиым болса, polyval функциясы дәрежелік көпмүше мәндерінің шамасын *x* жиым элементтерінің барлық мәндерін де кері қайтарады. *A* векторының коэффициенттерінің бар *c* көпмүшесінің түбірлерін есептеу roots(A) функциясы арқылы жүзеге асырылады:

```
»r=roots(h)
r =
     3.7656
    -0.26556
»X=-5:0.05:5;Y=polyval(h,X);plot(X,Y);hold on; scatter(r,polyval(h,r),40,'filled'); grid on
```

Алынған график:



Conv функциясы *w* коэффициентінің векторының көпмүшесін қайтарады және ол берілген коэффицеңттердің векторы *a* және *b* болатын екі көпмүшенің туындысы болып табылады, яғни w=conv(a,b)

Deconv функциясы *u* және *v* коэффициенттерінің векторының көпмүшесін қайтарады. Ол көпмүшеден қалған қалдық болып саналады. Коэффициенттің *a* және *b* векторлары арқылы *d* көпмүшесіне берілген. [u,v]=deconv(a,b). Мысалы:

```
» a=[4 5 8]; b=[2 3 4 7]; d=conv(a,b)
d =
   8   22   47   72   67   56
» deconv(d,b)
ans =
   4    5    8
» d(1)=d(1)+3
d =
  11   22   47   72   67   56
» deconv(d,b)
ans =
      5.5      2.75      8.375
» [x,y]=deconv(d,b)
x =
      5.5      2.75      8.375
y =
        0         0         0    -2.625     14.25    -2.625
```

Көпмүшелерді дифференциалдауды polyder функциясы орындайды. Бұл функцияның дәлелі дифференциалданатын көпмүше коэффициенттерінің векторы болып табылады. Функция, тиісінше, туынды көпмүшенің коэффициенттерінің векторын кері қайтарады.

```
» a=[4 5 7]; d= polyder(a)
d =
   8    5
»
```

MATLAB-та көпмүшені интегралдаудың да функциясы бар.

6.6. Аппроксимациялау

B=polyfit(x,y,n) функциясы n дәрежелі көпмүшенің *y(x)* кестелі функциясының аппроксимациясына пайдаланылады, ол дәрежелік көпмүшенің *B* векторын кері қайтарады. Функция параметрлері: *x, y* – аргумент пен функцияның векторлық мәні, *n* – көпмүшенің дәрежесі. Функцияны аппроксимациялаудың нәтижелерін *2, 3, 4* және *5* дәрежелік көпмүшелер арқылы қарастырамыз. Сценарийі:

```
x=1:1:10;
y=5+(0.1*sin(x));
scatter(x,y,50,'filled')

b2=polyfit(x,y,2)
b3=polyfit(x,y,3)
b4=polyfit(x,y,4)
b5=polyfit(x,y,5)

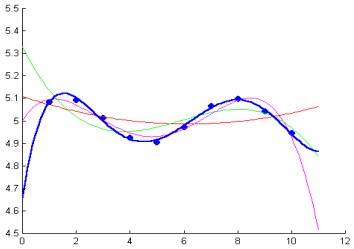
xp=0:0.1:11;
hold on
plot(xp,polyval(b2,xp),'r')
plot(xp,polyval(b3,xp),'g')
plot(xp,polyval(b4,xp),'m')
plot(xp,polyval(b5,xp),'b','LineWidth',2)
```

Нәтижесі:

```
» l7_p11
b2 =
     0.00323858317961304       -0.0397381205521077
5.10798609433367
b3 =
  Columns 1 through 3
    -0.00258158641063002       0.0458347589550084
-0.236196846401052
  Column 4
       5.32948620836573
b4 =
```

```
  Columns 1 through 3
    -0.000954614601775185        0.0184199348284241
-0.107858191930798
  Columns 4 through 5
       0.183833578380035         5.00186247703647
b5 =
  Columns 1 through 3
     0.000245186199226074      -0.00769723508049221
0.0862547832809714
  Columns 4 through 6
       -0.411276113473067        0.761819178688985
4.65124621214316
»
```

Графигі:



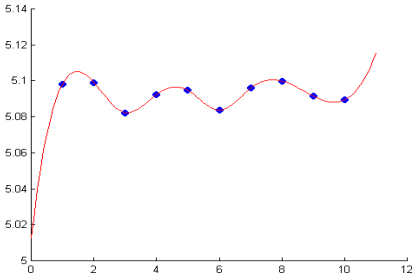
6.7. Интерполяциялау

Бір айнымалылы текше сплайндық кестелік (у)х функциясын интерполяциялау үшін yi=spline(x, y, xi) функциясы қолданылады. Мұндағы, *x, y* - аргумент пен функция мәні, *yi* функциясы *xi* аргументі үшін функция мәнін қайтарады. Мысалы:

```
% кестелі функцияны береміз:
x=1:1:10;
y=5+(0.1*sin(x).^0.1);
scatter(x,y,50,'filled')
hold on
```

```
xp=0:0.1:11;
% функция графигін құру үшін интерполяция сплайынын
пайдаланамыз:
plot(xp,spline(x,y,xp),'r')
```

Графигі:



yi=interp1(x,y,xi,'әдіс') функциясы *y(x)* кестелік функциясын интерполяциялау үшін қолданылады. Мұндағы, interp1 функциясы xi аргументіндегі yi кестелік функциясының мәнін кері қайтарады. Ол функцияны қолданғанда бірнеше әдіс қолданылуы мүмкін: 'nearest' – сатылық интерполяция; 'linear' - сызықтық интерполяция (егер әдіс көрсетілмесе, онда ол тұрақты орындалады); 'spline' - текше сплайндарымен интерполяция (сол сияқты spline функциясы); 'cubic' - текше интерполяциясы. Мысалы:

```
close all;
% кестелік функция береміз
x=1:1:10;
y=5+(0.1*sin(x).^0.1);
xp=0.5:0.1:11;

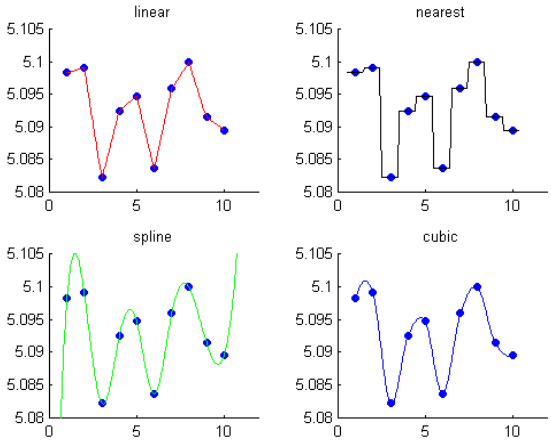
subplot(2,2,1)
scatter(x,y,30,'filled')
hold on
plot(xp,interp1(x,y,xp),'r')
title('linear')
axis([0,12,5.08,5.105])
```

```
subplot(2,2,2)
scatter(x,y,30,'filled')
hold on
plot(xp,interp1(x,y,xp,'nearest'),'k')
title('nearest')
axis([0,12,5.08,5.105])

subplot(2,2,3)
scatter(x,y,30,'filled')
hold on
plot(xp,interp1(x,y,xp,'spline'),'g')
title('spline')
axis([0,12,5.08,5.105])

subplot(2,2,4)
scatter(x,y,30,'filled')
hold on
plot(xp,interp1(x,y,xp,'cubic'),'b')
title('cubic')
axis([0,12,5.08,5.105])
```

Әртүрлі мысалдардан алынған графиктер:



Бақылау жұмысы. MATLAB-та теңдеулерді шешу.

Жұмыстың мақсаты:

1. MATLAB пакетінің негізгі жағдайларымен танысу.

2. Айнымалылармен, функциялармен және математикалық өрнектермен жұмыс.

3. Сызықтық емес теңдеулердің жүйесін шешу әдістерін игеру.

4. Сызықтық емес екі теңдеуден тұратын жүйені шешу әдістерін игеру.

5. Диалогтық режимде жұмыс істеудің тәжірибелік жолдарын игеру.

Жұмыстың тапсырмасы:

1 есеп. Сызықтық емес теңдеулерді шешу.

- f1(x) функциясы үшін M-функция құру.
- Программа файлын құру. Есептің тақырыбын комментарий ретінде енгізу.
- y(x)=f1(x) функциясын XY график ретінде шығару. Ол бойынша y(x)=0 теңдеуінің түбірін шамамен анықтау. Егер түбірлер графикте көрінбесе, онда аргументтің өзгеру аумағын өзгертіп, операцияны қайталау керек.
- Әр түбірге fzero функциясын қолданып, оның нақты мәнін табу.
- Нәтижелер жолын қалыптастырып, оны график терезесінің тақырыбына шығару.

2 есеп. Сызықтық емес екі теңдеуден тұратын жүйені шешу.

- f2(x) және f3(x) = f1(x) - f2(x) функциялары үшін M-функциялар құру.
- Программа файлын құру. Есептің тақырыбын комментарий ретінде енгізу.
- f1(x) және f2(x) функцияларын XY график ретінде шығару. Ол бойынша f1(x) және f2(x) графиктерінің қиылысы ретінде теңдеулер жүйесінің түбірін шамамен анықтау. Егер түбірлер графикте көрінбесе, онда аргументтің өзгеру аумағын өзгертіп, операцияны қайталау керек.
- Әр түбірге fzero функциясын қолданып, оның нақты мәнін табу.

▪ Нәтижелер жолын қалыптастырып, оны график терезесінің тақырыбына шығару.

Тапсырмалар варианты

№	Коэффициенті а болатын 3-ші дәрежелі f1(x)- полином				f2(x)
	a3	a2	a1	a0	
1	0	-1	4	-1	0.2exp(x)-20
2	0	2	-2	-15	40|cos(x)|
3	0	1	4	-1	10ln(x+5.5)
4	0	9	-8	-70	100|sin(x)|
5	0	-4	4	50	70cos(x)
6	.1	-5	4	40	60exp(|0.1*x|)-100
7	.2	-3	2	30	20sin(2x)
8	.3	-6	1	50	exp(|x|)sin(2x)
9	.4	-9	1	70	exp(|x|)cos(3x)
10	.5	-7	5	60	-60|cos(x)|
11	-.1	-4	9	60	15log(x+5.1)
12	-.2	-6	-7	55	-50ln(x+5.1)
13	-.3	-9	-8	75	-100|cos(x)|
14	-.4	7	8	-75	100sin(x/2)
15	-.5	1	4	-1	40cos(x/2)

Орындау мысалдары.

Тапсырма

1-ші функция f1(x) = -0.85*x3-2x2+7x+2

2-ші функция f2(x) = 6cos(x) - 5

Өрнектердің түбірлерін табу үшін керекті өрнектерді M-функциялар ретінде жазып, MATLAB-тың процедураларын қолданамыз. MATLAB редакторында мәтіндерін fun1.m, fun2.m және fun3.m файлдарында сақтайтын fun1, fun2 және fun3 жаңадан кірістірілген M-функцияларын құрамыз. Үшінші функция екі теңдеуден тұратын жүйелерді, сол алгоритм бойынша екі теңдеулердің айырмасы ретінде өрнектеу арқылы, шешу үшін керек.

fun1.m файлы function f1=fun1(x)

f1=-0.85*x.^3-2*x.^2+7*x+2

fun2.m файлы function f2=fun2(x)

```
f2=6*cos(x)-5
```
fun3.m файлы
```
function f3=fun3(x)
f3=-0.85*x.^3-2*x.^2+7*x+2-6*cos(x)+5
```
1 есеп
```
% 1Есеп
% Өрнектің түбірлерін табу
% Аргументтің шектері мен адымы
a=-4;
b=4;
h=0.5;
% Аргумент векторы
x=[a:h:b];
% Түбірлердің оқшау бөлігінің графигі
plot(x,fun1(x));grid on;
% Бірінші түбірді табу
x1=fzero('fun1(x)',[-4 -3]);
% Екінші түбірді табу
x2=fzero('fun1(x)',[-1 0]);
% Үшінші түбірді табу
x3=fzero('fun1(x)',[1 3]);
% Нәтижелер жолын алу
Result=strcat('x1=',num2str(x1),'   x2=',num2str(x2),'
x3=',num2str(x3));
% Оны графикке тақырып ретінде қосу
title(Result)
```

2 есеп
```
% 2 Есеп
% Сызықтық емес теңдеулер жүйесін шешу
% Аргументтің шектері мен адымы
a=-4;
b=4;
h=0.5;
% Аргумент векторы
x=[a:h:b];
% Түбірлердің оқшау бөлігінің графигі
plot(x,fun1(x),x,fun2(x));grid on;
```

```
% Бірінші түбірді табу
x1=fzero('fun3(x)',[-4 -3])
% Екінші түбірді табу
x2=fzero('fun3(x)',[-1 0]);
% Үшінші түбірді табу
x3=fzero('fun3(x)',[2 3]);
% Нәтижелер жолын алу
Result=strcat('x1=',num2str(x1),'   x2=',num2str(x2),'
x3=',num2str(x3));
% Оны графикке тақырып ретінде қосу
title(Result)
```

Бақылау сұрақтары

1. Апостроф және sym функциясы арқылы символдық айнымалыларды беру.
2. Кеңейту функциясы (дәрежелері бойынша орналастыру) - expand.
3. Жинақтау функциясы (қысқарту) - simple.
4. Қысқарту функциясы - simplify.
5. Факторлау функциясы (қарапайым көбейткіштерге жіктеу) - factor.
6. Дәреже бойынша жинау функциясы - collect.
7. Ішкі өрнектің айнымалысын алмастыру функциясы - subexpr.
8. Интегралдау функциясы - int.
9. Символдық формадағы теңдеулерді шешу функциясы - solver.

7. MATLAB-ТАҒЫ ДИФФЕРЕНЦИАЛДЫҚ ТЕҢДЕУЛЕР ЖҮЙЕСІ

Дифференциалдық теңдеулер – бұл бір немесе бірнеше айнымалылардан тұратын белгісіз функциясы бар және тәуелсіз айнымалылары бар белгісіз функциясының туындысы бар теңдеу. Бір айнымалысы бар дифференциалдық теңдеудің түрі келесідей болады.

$$\frac{dy}{dt} = f(t, y).$$

Ал n-ші ретті дифференциалдық теңдеулер жүйесі

$$\begin{cases} y_1' = f_1(t, y_1, y_2 \ldots y_n) \\ y_2' = f_2(t, y_1, y_2 \ldots y_n), \\ \quad \ldots \\ y_n' = f_n(t, y_1, y_2 \ldots y_n) \end{cases}$$

мұндағы t - тәуелсіз айнымалы, тәртіп бойынша – бұл уақыт.

Дифференциалдық теңдеулер жүйесінің дербес шешімін (Коши есебі) табу үшін бастапқы шарттар: $y_1(t_0)$, $y_2(t_0)$, $y_3(t_0)$, $\ldots y_n(t_0)$ берілуі тиіс. Дифференциалдық теңдеулер жүйесінің алғашқы шарттары бойынша шешімі $y_1(t)$, $y_2(t)$, $y_3(t)$, $\ldots y_n(t)$ тәуелділіктеріне алып келеді.

Әрбір n-ші ретті айқын түрде берілген дифференциалдық теңдеулерді $y^{(n)}(t) = f(t, y, y', y'', \ldots y^{(n-1)})$ жаңа белгісіз $y_1 = y$, $y_2 = y'$, $y_3 = y''$, $\ldots y_n = y^{(n-1)}$ функцияларын енгізу арқылы n дифференциалдық теңдеулер жүйесіне түрлендіруге болады:

$$\begin{cases} y_1' = y_2 \\ y_2' = y_3 \\ \dots \\ y_{n-1}' = y_n \\ y_n' = f(t, y_1, y_2, \dots, y_n) \end{cases}.$$

MATLAB-та дифференциалдық теңдеулер жүйесін сандық жолмен шешу үшін ode45, ode23, ode113, ode15s, ode23s, ode23t және ode23tb функциялары қолданылады. Бұл функциялар түрлі сандық әдістерді қолданады, мысалы, ode45 - Рунге-Куттаның 4- және 5- реттік әдістері, ode23 - Рунге-Куттаның 2- және 3-реттік әдістері. Ал ode15s, ode23s, ode23t, ode23tb функциялары дифференциалдық теңдеулердің қатты жүйелерін шешуде қолданылады. Бұл дифференциалдық теңдеулердің шешімдері тәуелсіз айнымалылардың түрлі аралықтарында әр түрлі болады, яғни біріншісінде өте жоғары ауытқулар байқалса, келесісінде өте төмен болады.

Дифференциалдық теңдеулердің жүйелерін шешуде ode45 функциясының қолданылуын келесідей мысалдан көруге болады. $Y = [y_1, y_2, \dots y_n]$ және $F = [f_1, f_2, \dots f_n]$ векторлары енгізілсін. Онда дифференциалдық теңдеулер жүйесі $Y' = F(t, Y)$ түрге ие болады.

Дифференциалдық теңдеулер жүйесін шешетін функцияны қолдану үшін ($F(t,Y)$) теңдеулер жүйесінің оң жақ бөлігін шешетін арнайы m-функциясын құру керек.

Функцияны шақыру үшін [t,Y]=ode45('функция атауы',ts,Y0) қолданамыз. Ode45 функциясы дифференциалдық теңдеулер жүйесін ts=[tn, tk] уақыт аралығында *Y0* бастапқы шартымен интегралдайды. Мұндағы бірінші параметр – ол дифференциалдық теңдеулер жүйесінің оң жағын есептейтін *m* -функцияның атауы. Кері қайтарылатын *Y* жиымының әр жолы *t* векторы арқылы анықталатын уақыттағы шешімі болады (бірінші бағанда – y_1, ал екіншісінде - y_2 және т.т.).

Мысалы. Ode45 және ode15s функцияларын қолданып,

$$\begin{cases} \dfrac{dy_1}{dt} = 5y_2 * \sin(t) \\ \dfrac{dy_2}{dt} = t + 1 - y_1 - t \cdot y_2 \end{cases}$$

түріндегі дифференциалдық теңдеулер жүйесін шешу керек. Алынған тәуелділік графигін әрбір терезеге жекелеп тұрғызайық.

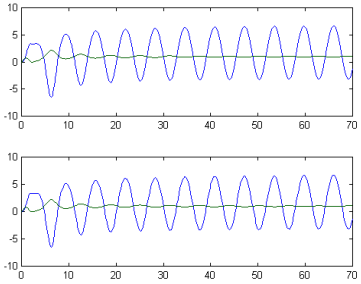
Дифференциалдық теңдеулер жүйесінің оң жақ бөлігін шешуге арналған m-функциясы:

```
function Y=l7_p7f(t,y)
Y=[ 5*y(2)*sin(t);t+1-y(1)-t*y(2) ];
```

Сценарий мәтіні:

```
y0=[0,0];
[T,Y]=ode45('l7_p7f',[0,70],y0);
subplot(2,1,1)
plot(T,Y);
[T1,Y1]=ode15s('l7_p7f',[0,70],y0);
subplot(2,1,2)
plot(T1,Y1);
```

Алынған графиктер:



Бақылау жұмысы.

MATLAB-та символдық есептеулер.

Жұмыстың мақсаты:

1. Символдық есептеу пакетінің негізгі жағдайларымен танысу.

2. Символдық айнымалылармен, матрицалармен және математикалық өрнектермен жұмыс.

3. Символдық аналитикалық есептеулерді, яғни өрнектерді қысқарту, алгебралық теңдеулерді шешу, сызықтық теңдеулер жүйесін шешу, қатарлардың қосындыларын есептеуді игеру.

4. Символдық интегралдау мен символдық дифференциалдауды игеру.

5. Диалогтық режимде жұмыс істеудің тәжірибелік жолдарын игеру.

Жұмыстың тапсырмасы

1 есеп. Ашу/жинақтау.

- f1(x) өрнегін енгізіп, оны ашу.
- Алынған өрнекті жинақтау. Нәтижені f1(x)-пен салыстыру.

2 есеп. Дифференциалдау/интегралдау.

- f1(x) өрнегін енгізіп, x бойынша туындысын табу.
- Алынған өрнек үшін анықталмаған интеграл табу. Нәтижесін f1(x)-пен салыстыру.

Тапсырмалар варианты

№	f1(x)	f2(x)
1	$(1+x)^2$	ax^3+bx^2+cx+d
2	$(1-x)^2$	sin(ax)
3	$(a+x)^2$	cos(ax)
4	$(a-x)^2$	sec(x)
5	$(1+x)^3$	exp(ax)
6	(1-x)3	x(ln(x)-1)
7	(a+x)3	-csc(x)
8	(a-x)3	$1/(1+x^2)$
9	$(1+x)^4$	1/(a+bx)
10	$(1-x)^4$	$1/(1-x^2)$
11	$(a+x)^4$	$-\cos^3(x)/3$
12	$(a-x)^4$	$\sin^3(x)/3$
13	$(1+x)^5$	$x^2(\ln(x)-0.5)/2$
14	$(1-x)^5$	-(ln(x)+1)/x
15	$(a+x)^5$	$\ln^2(x)/2$
16	$(a+x)^5$	$\ln^3(x)/3$

Орындау мысалдары.

Тапсырма $f1(x)=(a+x)^3$

$f2(x)=\sin(x)/x$

```
1 есеп
% 1 есеп
% Символдық айнымалыларды анықтау
syms a x;
% Функция
f1=(a+x)^3
% Оны кеңейту
f1_new=expand(f1)
% Кеңейтілгендерді жинақтау
f1_old=simple(f1_new)
2 есеп
% 2 есеп
% Символдық айнымалыларды анықтау
syms a x;
% Функция
f1=(a+x)^3
% x бойынша алғашқы туындыны табу
f1_new=diff(f1,'x',1)
% x бойынша анықталмаған интеграл табу
f1_old=int(f1_new,'x')
```

Бақылау сұрақтары

1. ode45 функциясының міндеті.
2. fun функциясының міндеті.
3. fzero функциясының міндеті.
4. diff(x) функциясының міндеті.
5. trapz(x,y) функциясының міндеті.
6. cumtraps (x,y) функциясының міндеті
7. quad және quad8 функцияларының міндеттері.

ҚОЛДАНЫЛҒАН ТЕРМИНДЕРДІҢ ҚЫСҚАША ОРЫСША-ҚАЗАҚША СӨЗДІГІ

А
Активный – белсенді
Апостроф – дәйекше
Б
Библиотека – кітапхана
В
Вариант – нұсқа
Ввод данных – деректерді енгізу, беру
Ввод текста – мәтінді енгізу
Ветвь – тармақ
Вкладка –қосымша бет, қыстырма
Включатель – өшіріп-қосқыш
Вставка – кірістіру
Встроенный – кіріктірілген
Входной сигнал – кіріс сигналы
Вызов – шақыру
Выходной сигнал – шығыс сигналы
Вязкость – тұтқырлық
Г
Глобальные переменные – аумақты айнымалылар
Глобальный – аумақты
Группировка – топтастыру
Д
Данные – берілген деректер
Действие – іс-қимыл
Действительный - нақты
Диалог – сұхбат
Доступ – қолжетімдік дәреже, ену, рұқсат
З
Знак – таңба
Значение – мән, мағына
И
Изоляция – оқшаулау
Инвертирование – кері таңбалау

Интервал – аралық
Информация – ақпарат
Источник сигнала – сигнал көзі
К
Клавиша – перне
Ключ – кілт
Ключевое слово – түйінді сөз
Кнопка – батырма
Колебательное звено – тербелмелі буын
Комментарий – түсініктеме
Комплексно-сопряженный – комплекстік-түйіндес
Конечная разность – шектік айырым
Константа – тұрақты
Крутизна – құламалық, құлдилық
Курсор – мегзегіш
Л
Легенда – әфсана легенда (картаға, сызбаға берілген түсіндірме сөздер)
Локальный – жергілікті
М
Массив – жиым
Меню – мәзір
Метод – әдіс, тәсіл
Многочлен – көпмүше
Н
Наглядность – көрнекілік
Настройка – күйге келтіру
О
Область – аймақ
Обозреватель – шолушы
Окно команд – командалар терезесі
Окно подсказок – еске салу терезесі
Операция – іс-әрекет, амал
Отладчик – дұрыстаушы
Отношение – қатынас
Отображение – көріну, кейіптеу
Отрицание – терістеу
Ошибка – қате

П
Палитра – түс түрлері
Память – жад
Панель инструментов – құралдар жақтауы
Передаточная функция – беріліс функциясы
Переменная – айнымалы
Пересечение – қиылысу
Подинтегральная величина – интеграласты өрнек
Подсистемы – ішкі жүйелер
Поле – өріс
Пользователь – қолданушы
Преобразование – түрлендіру
Приоритет – басымдық
Присвоение – меншіктеу
Проблемные ситуации – ахуалды мәселелер
Программа – бағдарлама
Пространство – кеңістік
Процент – пайыз
Процесс – үдеріс
Путь поиска – іздеу жолы
Р
Рабочее пространство – жұмыс кеңістігі, аймағы
Регистрирующие блоки – тіркеуші блоктар
Регулятор – реттегіш
С
Сброс – бастапқы күйге түсіру
Сдвиг – жылжу
Система – жүйе
Сообщение – мәлімет
Состояние – күй
Спираль – шиыршық
Ссылка – сілтеме
Столбец – баған
Стрелка – бағдар
Строка – жол
Строка состояния – күй жолы
Структура – құрылым

Ступенчатое воздействие – сатылық әсер
Сумматор – жиынтықтаушы
Сухой – құрғақ
Схема – сұлба
Счетчик – санағыш
Т
Таблица – кесте
Текст – мәтін
Текущий – ағымдық
Тип данных – деректер типі
Точка останова – тоқтау нүктесі
Транспонирование – тасымалдау
У
Узловая точка – түйіндік нүкте
Управление – басқару
Установка – орнату, қондыру, қою
Ф
Флажок –жалауша
Формат – қалып
Формирование – қалыптасу
Фронт – шеп
Ш
Шаг – адым, қадам
Я
Ячейка – ұяшық

БИБЛИОГРАФИЯЛЫҚ ТІЗІМ

1. *Алексеев Е.Р., Чеснокова О.В.* MATLAB 7. Самоучитель. "НТ Пресс", 2006.

2. *Ануфриев И., Смирнов А., Смирнов Е.* MATLAB 7. Наиболее полное руководство. – СПб.: БХВ-Петербург, 2005.

3. *Бенькович Е.С.* и др. Практическое моделирование динамических систем. СПб.: БХВ-Петербург, 2002.

4. *В. Кондрашов, С.Королев.* Matlab как система программирования научно-технических расчетов. Мир. 2002.

5. *Веремей Е., Корчанов В., Коровкин М., Погожев С.* Компьютерное моделирование систем управления движением морских подвижных объектов. НИИ Химии СПбГУ. 2002.

6. *Гандер В., Гржебичек И.* Решение задач в научных вычислениях с применением Maple и MATLAB. "Вассамедина" 2005.

7. *Говорухин В., Цибулин В.* Компьютер в математическом исследовании: Maple, MATLAB, LaTeX. Питер. 2001.

8. *Гонсалес Р., Вудс Р., Эддинс С.* Цифровая обработка изображений в среде MATLAB. Москва: Техносфера, 2006.

9. *Гультяев А.* Визуальное моделирование в среде Matlab: Учебный курс. Питер. 2000.

10. *Дьяконов В.П., Абраменкова И., Круглов В.* MATLAB с пакетами расширений. Нолидж. 2001.

11. *Дьяконов В.П., Круглов В.* MATLAB. Анализ, идентификация и моделирование систем. Специальный справочник. Питер, 2001.

12. *Дьяконов В.П., Круглов В*. Математические пакеты расширения MATLAB. Специальный справочник. Питер, 2001.

13. *Иглин С.П.* Математические расчеты на базе Matlab. Издательство "BHV-Санкт-Петербург", 2005.

14. *Карлащук В. И.* Электронная лаборатория на IBM PC. Лабораторный практикум на базе Electronics Workbench и MATLAB. СОЛОН-Пресс, 2004.

15. *Кривилев А.* Основы компьютерной математики с использованием системы MATLAB. Лекс-Книга, 2005.

16. *Курбатова Е.А.* Matlab 7. Самоучитель. – М.: Издательский дом «Вильямс», 2006.

17. *Лазарев Ю.Ф.* Начала программирования в среде MatLAB: Учебное пособие. - К.: НТУУ "КПИ", 2003.

18. *Лазарев, Ю.* Моделирование процессов и систем в MatLab: учебный курс / Ю. Лазарев. – СПб.: Питер; Киев: BHV, 2005.

19. *Мартынов Н., Иванов А.*. Matlab 5.x. Вычисления, визуализация, программирование. Кудиц-образ. 2000.

20. *Мартынов Н.Н., Иванов А.* Matlab 5.x. Пособие по программированию в системе MATLAB. МГУ. 2000.

21. *Мартынов Н.Н.*. Введение в MATLAB 6. Кудиц-образ. 2002.

22. *Мэтьюз Д.и Финк К.* Численные методы. Использование MATLAB. Вильямс. 2001.

23. *Поршнев С.* Компьютерное моделирование физических процессов в пакете MATLAB. Горячая Линия - Телеком. 2003.

24. *Поршнев С.В.* MATLAB 7. Основы работы и программирования. "Бином. Лаборатория знаний" 2006.

25. *Потемкин В.* Вычисления в среде MATLAB. Диалог-МИФИ. 2004.

26. *Чен К., Джиблин П., Ирвинг А.* MATLAB в математических исследованиях. Мир. 2001.

27. *Штовба С.Д.* Проектирование нечетких систем средствами MATLAB.: Горячая Линия - Телеком, 2007.

28. *Өжікенов К.Ә.* Жүйелерді модельдеудің бағдарламалық құралдары (MATLAB/Simulink): Оқулық / *Өжікенов Қ. Ә.;* ҚР білім ж-е ғылым мин-гі.- Алматы: Дәуір, 2012. - 304 б.

29. *Ожикенов К.А.* Основы программирования в среде MATLAB: Учебное пособие. – Алматы: 2012. – 148 с.

МАЗМҰНЫ

	КІРІСПЕ……	3
1.	MATLAB ОРТАСЫ. КОМАНДАЛЫҚ РЕЖИМДЕГІ ЖҰМЫС.....	5
1.1.	MATLAB ортасы.....	5
1.2.	Командалық режимдегі жұмыс.....	6
1.3.	Математикалық өрнектер.....	9
1.3.1.	Нақты сандар.....	9
1.3.2.	Комплекстік сандар.....	9
1.3.3.	Айнымалылар.....	10
1.3.4.	Жүйелік айнымалылар.....	11
1.3.5.	Арифметикалық амалдар.....	12
1.3.6.	Қатынас амалдары.....	13
1.3.7.	Логикалық амалдар.....	13
1.3.8.	Элементар функциялар.....	14
1.4.	Жұмыс аймағы.....	16
1.5.	Жүйе параметрлерін күйге келтіру.....	19
1.6.	Анықтама жүйесін қолдану.....	21
2.	ЖИЫМДАРМЕН ЖҰМЫС.....	25
2.1.	Векторлар мен матрицалардың қалыптасуы	25
2.2.	Жиым элементіне сілтеме.....	28
2.3.	«Қос нүкте» операторын қолдану.....	29
2.4.	Берілген жиымдағы жолды және бағанды өшіру.....	31
2.5	Векторлар мен матрицаларға элементар функцияларды қолдану.....	32
2.6.	Матрицалық және векторлық амалдар.....	32
2.7.	Жиымдарға элементтері бойынша орындалатын амалдар.....	36
2.8.	Жиыммен жұмыс істеу функциялары.....	38
2.9.	Сызықтық теңдеулер жүйелерін шешу.....	44
2.10.	Жиымдарға қатынас және логикалық амалдарды қолданудың ерекшеліктері.....	46
2.11.	Амалдарды орындаудағы басымдықтар.....	47
2.12.	Матрицалармен жұмыс істеудің мысалдары.	48

3.	М-ФАЙЛДАР: СЦЕНАРИЙЛЕР ЖӘНЕ ФУНКЦИЯЛАР	53
3.1.	Сценарий	54
3.2.	Іздеу жолының қондырғысы	55
3.3.	Функциялар	58
3.4.	Аумақты айнымалылар	62
3.5.	М-файлдың редактор/дұрыстауы. Бағдарламаны дұрыстау	63
4.	ГРАФИК ТҰРҒЫЗУ	67
4.1.	Екі өлшемдік графиктерді тұрғызу	67
4.1.1.	Декарттық координат жүйесінде график тұрғызу	67
4.1.2.	Бір терезеде бірнеше график тұрғызу	72
4.1.3.	Полярлық координаттар жүйесінде график тұрғызу	72
4.2.	Графиктік терезелермен жұмыс	73
4.3.	Графиктік терезелердің шағын терезелерге бөлінуі	73
4.4.	Үш өлшемді графиктің тұрғызылуы	74
4.5.	Графиктік терезеге жазулар, тақырыптар және координаталық кесте шығару	80
4.6.	График масштабын өзгерту	81
4.7.	Арнайы график	81
4.8.	Қосымша мысалдар	87
5.	БАҒДАРЛАМАЛАУ	95
5.1.	Функцияны енгізу және шығару	95
5.2.	Циклдік операторлар	97
5.2.1.	For циклдік операторы	97
5.2.2.	While циклдік операторы	99
5.3.	Тармақталған оператор	100
5.3.1.	Шартты оператор	100
5.3.2.	Ауыстырып-қосқыш оператор	102
5.4.	Басқа операторлар	105
5.5.	Мысалдар	107
6.	MATLAB-ТАҒЫ ТИПТІК ЕСЕПТЕР	114
6.1.	Сызықтық емес теңдеулерді және теңдеулер жүйесінің шешу	114

6.2. Бір немесе бірнеше айнымалысы бар функциялардың минимумын анықтау..... 117
6.3. Сандық дифференциалдау мен шектік айырымды есептеу..... 120
6.4. Сандық интегралдау..... 122
6.5. Көпмүшелерге қатысты есептерді шығару.... 125
6.6. Аппроксимациялау..... 128
6.7. Интерполяциялау..... 129
7. MATLAB-ТАҒЫ ДИФФЕРЕНЦИАЛДЫҚ ТЕҢДЕУЛЕР ЖҮЙЕСІ..... 136
Библиографиялық тізім..... 145

Оқулық басылым

Қасымбек Әділбекұлы Өжікенов
Раузия Сыдыққызы Исмагулова

MATLAB МАТЕМАТИКАЛЫҚ ОРТАСЫ

Оқу құралы

ОБО РБ бастығы *З. А. Ғұбайдулина*
Редакторы *А.И. Бейсебаева*
Компьютерде беттеген *С.Е. Орысбекова*

Басуға қол қойылды 20.05.2016 ж.
Таралымы 300 дана. Пішімі 60х84х 1/16.
№ 1 баспаханалық қағаз.
Е.-б.т 8.8., Ш.б.т. 9.5. Тапсырыс № 738. Бағасы келісімді.

Қ. И. Сәтбаев атындағы Қазақ ұлттық техникалық
зерттеу университетінің басылымы,
Оқу-баспа орталығы,
Алматы қ., Сәтбаев көшесі, 22

ISBN 978-601-228-946-6

