ҚАЗАҚСТАН РЕСПУБЛИКАСЫ БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ

Қ.И.Сәтбаев атындағы Қазақ ұлттық техникалық университеті

Ақпараттық және телекоммуникациялық технологиялар институты
«Ақпараттық технологиялар» кафедрасы

В.З.Абдуллина, Ә.Ө. Мұртазина

ИНЖЕНЕРЛІК ЕСЕПТЕУЛЕРДІҢ САНДЫҚ ӘДІСТЕРІ

«Бизнес қосымшалардағы сандық әдістер» пәні бойынша студенттердің өзіндік жұмыстарына арналған әдістемелік нұсқаулар
5В070300 – «Ақпараттық жүйелер» мамандығының студенттері үшін

Алматы 2015

ӘОК 681.3.06

ҚҰРАСТЫРУШЫЛАР: В.З.Абдуллина, Ә.Ө. Мұртазина. Инженерлік есептеулердің сандық әдістері. 5В070300 – «Ақпараттық жүйелер» мамандығының студенттері үшін «Бизнес қосымшалардағы сандық әдістер» пәні бойынша студенттердің өзіндік жұмыстарына арналған әдістемелік нұсқаулар. – Алматы: ҚазҰТУ, 2015. – 34 б.

«Бизнес қосымшалардағы сандық әдістер» пәні бойынша «Инженерлік есептеулердің сандық әдістері» студенттердің өзіндік жұмыстарына арналған әдістемелік нұсқаулар 5В070300 – «Ақпараттық жүйелер» мамандығының студенттеріне арналған. Әдістемелік нұсқаулар «Бизнес қосымшалардағы сандық әдістер» пәні бойынша студенттердің өзіндік жұмыстарын орындау үшін мысалдар, жеке тапсырмалардан тұрады.

Пікір беруші: Дадаева И.Г. - техн. ғыл. канд., доценті

КІРІСПЕ

«Инженерлік есептеулердің сандық әдістері» әдістемелік нұсқауы 5В070300 - «Ақпараттық жүйелер» мамандығының студенттеріне «Бизнес қосымшалардағы сандық әдістер» пәні бойынша өзіндік жұмыстарын жасауға арналған. «Бизнес қосымшалардағы сандық әдістер» курсы бизнес қосымшалардағы компьютер көмегі арқылы математикалық және инженерлік есептеулерді орындау әдістерін және принциптерін игеруге, компьютерлік математика программалау пакеттеріндегі талдау және алгебра сандық әдістерінен білім алып, дағдылану және оларды ғылыми-техникалық есептеулер мен бизнес қосымшалардағы зерттеулерге қолдануға мүмкіндік береді.

Әмбебап матрицалық формада берілген кең ауқымды математикалық, ғылыми және инженерлік есептерді шешуге арналған MATLAB пакеті ресми қабылданған және компьютерлік математиканың сенімді жүйесі болып табылады [1-10]. Аэроғарыштық саласында, әскери-өндірістік кешен жүйесінде, автомобиль жасауда, экономика және бизнес және т.б. салаларда белсенді қолданылатын, көп стандартты пакеттері және қосалқы программаларымен техникалық есептеулерге арналған жоғарғы дәрежелі программалау тілін нарықта үздік MATLAB жүйесі ұсынады. Алдыңғы қатарлы даму тәжірибесін және сандық әдістердің компьютерлік іске асырылуын MATLAB пакеті өзіне сіңірген.

Қиын инженерлік және математикалық есептеулерді орындау шешу мүмкіндігі бар, қазіргі кездегі есептеу пакеттерін қолдануға дағдылану және білімді бекітіп, терендетуге «Бизнес қосымшалардағы сандық әдістер» пәні бойынша көмектесетін студенттін өзіндік жұмыстарын орындау нұсқаулары ретінде сандық әдістер бойынша есептерді шешу әдістерінің сипаттамасы әдістемелік нұсқауда баяндалған.

1. MATLAB-тағы САНДЫҚ ӘДІСТЕР

1.1Сызықты теңдеулер жүйесін

Сызықты теңдеулер жүйесін шешуде матрицалық әдістер кең қолданылады. Сызықты теңдеулер жүйесі келесі түрде болады:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n &= b_2 \\ &\ \ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n &= b_n \end{aligned} \qquad (1)$$

мұндағы, $a_{11}, a_{12},..., a_{nn}$ – нақты немесе комплексті мәндерге ие болатын A матрицасының элементтері, $x_1, x_2,..., x_n$ – X векторын құрайтын белгісіздер және $b_1, b_2,..., b_n$ – B векторын құрайтын бос мүшелер (нақты немесе коплекстік). Бұл жүйе **AX=B** матрицалық түрде берілуі мүмкін, мұндағы A – белгісіздері бар коэффицциенттердің матрицасы, X – белгісіздердің ізделінетін векторы және B – бос мүшелердің векторы.

Сызықты теңдеулер жүйесін шешу үшін келесі әдістер қолданылады: Гаусс әдісі, Крамер әдісі, матрицалық әдіс.

Әрбір белгісізі анықтаушылардың қатынасы түрінде берілетін Крамер әдісін қарастырайық. *N*-ші ретті матрицаның анықтауышы деп жүйенің коэффициенттерінен құралған *D* санын айтамыз:

$$D= \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} = \sum(-1)^k a_{1\alpha}a_{2\beta}\dots a_{n\omega}. \qquad (2)$$

D_j – анықтауыш, ол *D*-дан x_j белгісіздерінің a_{kj} коэффициенттерінен тұратын бағанды b_k бос мүшелердің бағанымен ауыстырғанда алынады. Мысалы:

$$D_1= \begin{vmatrix} b_1 & a_{12} & \dots & a_{1n} \\ \dots & \dots & \dots & \dots \\ b_n & a_{n2} & \dots & a_{nn} \end{vmatrix}$$

Егер жүйенің анықтауышы $D \neq 0$ болса, онда (1) жүйе анықталған және бір ғана шешімі бар. Жүйенің түбірлері Крамердің формуласы арқылы табылады:

$$X_1= D_1/D, \quad x_2= D_2/D, \ \dots, \ x_n=D_n/D.$$

Егер $D = 0$ болса, онда (1) жүйесі байланыссыз және шешімі жоқ. Айнымалы ретті сызықты теңдеулер жүйесін шешу үшін Крамердің ережесін қолдануға тырысып көру керек. Бірақ теңдеулердің саны көп болса, оған n белгісіздерді есептеу үшін $n+1$ анықтауышты есептеу керек болады, сондықтан да өте көп арифметикалық операциялар орындау керек болады. (2) өрнектен анықтауыш $n!$ қосындысына тең екендігін аңғаруға болады, олардың әрқайсысы n элементінің көбейтіндісі. Сондықтан да n ретті анықтауыш үшін $(n+1)n!$ көбейту және $n!-1$ қосу қажет, яғни арифметикалық операциялардың жалпы саны келесі өрнекке тең:

$$N=(n+1)(n*n!-1)+n \quad .$$

Сондықтан да Крамер ережесін бірнеше теңдеулерден тұратын жүйелерді шешу үшін қолдана аламыз.

A матрицасының түріне және оның сипаттамалық ерекшеліктеріне сай MATLAB-та шешудің әртүрлі әдістерін қолдануға болады.

MATLAB-та СТЖ шешудің үш қарапайым тәсілін қарастырайық:

1) X=B/A;
2) X=B*A^(-1);
3) X=B*inv(A).

1мысал. Теңдеулер жүйесі берілген:

$4x_1 + 5x_2 + 6x_3 = 31$

$4x_1 + 4x_2 + 5x_3 = 37$

$3x_1 + 3x_2 + 4x_3 = 41$

А және В матрицаларын береміз:

A = [4 4 3; 5 4 3; 6 5 4]; B = [31 37 41];

1) X1=B/A; $\rightarrow$ X1 = [2.0000 -59.0000 53.0000];

2) X2=B*A^(-1); $\rightarrow$ X2 = [2.0000 -59.0000 53.0000];

3) X3=B*inv(A); $\rightarrow$ X3 = [2.0000 -59.0000 53.0000].

Сызықты теңдеулер жүйесінің басқадай шешу әдістерін қарастырамыз.

1) AX=B сызықтық теңдеулер жүесінің (ряда) шешімін X=B/A өрнегі береді. Мұндағы A матрицасының өлшемі mxn және B матрицасының өлшемі nxk.

2) XA=B сызықтық теңдеулер жүйесінің шешімін X=B/A өрнегі береді. Мұндағы A матрицасының өлшемі mxn және B матрицасының өлшемі nxk. Егер де А квадратты матрица болса, мысалға A/B да сол сияқты, inv(A)* B да, басқа жағдайда төмендегідей варианттарды қарастырамыз.

3)Егер A матрицасының өлшемі mxn, ал n компоненттері бар B вектор бағаны немесе бірнеше ұқсас бағандардан құралған матрица болса сонда AX=B өрнегінің (теңдеуінің) шешімі X=A/B, ол белгілі Гаусс әдісімен шешіледі.

MATLAB – та шегі бар СТЖ мен сиректелген матрицалары бар СТЖ-ні шешуге арналған көптеген функциялар бар. Оларға кіші квадраттар әдісі, түйіскен градиенттер әдісі (екібағытты, итерациялық, квадраттық әдістер) және ортақ байланыстың жоқтығын минималдаудың итерациялық әдісі қолданылады.

Шектері бар сызықты теңдеулер жүйесін кіші квадраттар әдісі арқылы шешуге арналған кейбір функцияларды қарастырайық:

1) X =lscov(A,B,V) - *A*X=B + e* түріндегі СТЖ-нің шешімін X векторы ретінде қайтарады, мұндағы *e – V* ковариациялық матрицасын беретін шу векторы; кіші квадраттар әдісі ковариациясы белгілі шулардың қатысымен іске асырылған; тік бұрышты *A* матрицасының өлшемі $m\times n$ болу керек, мұндағы $m > n$; шешу барысында келесі өрнек минималданады: *(AX-B)'*inv(V)*(AX - B)*; шешімнің түрі *X=inv(A'* inv(V)*A)*A'*inv(V)*B* өрнегіндей болады, бірақ шешу алгоритмі *V* матрицасының инвертирлеу операциясын жүргізбей құрылған;

2) [X, dX] = lscov(A,B,V) – *X* стандартты қателігін *dX* айнымалысы түрінде қайтарады;

3) X =isqnonneg(A,B) - *AX=B* СТЖ-сін оң шектері бар кіші квадраттар әдісі арқылы шешу, *A* – нақты тікбұрышты матрица, *B* – нақты вектор.

Сиректелген матрицалары бар СТЖ-ні шешу - кесімді матрицалар аппаратын қолдану сфераларының бірі. Төменде сиректелген матрицаларды қолдану облысына қатысты функциялар келтірілген. Сипатталған әдістердің көбісі итерациялыққа жатады, яғни олардың шешімі берілген қателікте немесе максималды ақиқатқа жақын нәтижесін алуға апаратын қадамдар – итерациялар барысында алынады. MATLAB-тың төменде сипатталған функциялары қарапайым СТЖ-ні (сиректелген матрицалары жоқ) шешуге де қолданылады:

1) lsqr(A, B) - егер матрица тізбектелген болса, оның $A*X=B$ СТЖ-сінің X нақты шешімін қайтарады, қарсы жағдайда – кіші квадраттар итерациялық әдісімен алынған шешімді қайтарады. A коэффициенттерінің матрицасы $m \times n$ өлшемді тікбұрышты болуы қажет, ал B теңдігінің оң жақ бөлігінің вектор-бағанасының өлшемі m-ға тең. $m >= n$ шарты міндетті болмауы да мүмкін. *Lsqr* функциясы итерацияны алғашқы бағадан бастайды, келісім бойынша нөлдерден тұратын n өлшемді вектордан тұрады. Итерация шешімге ұқсас болғанша, немесе қате пайда болғанша, немесе итерацияның максималды санына жеткенше (келісім бойынша *min(20, m, n)* тең - 20-ға немесе тендеулердің немесе белгісіздердің санына тең) жүргізіледі. Ұқсастық векторларының екінші нормасының қатынасы *norm(B - AX)/norm(B) <= tol* болған жағдайда іске асады (келісім бойынша 1e-6).

2) lsqr(A, B, tol) – *tol* берілген қателігі бар (таңдап алу арқылы) шешімді қайтарады.

3) lsqr(A, b, tol, maxit) – келісім бойынша берілген өте кіші санның орнына *maxit* берілген максималды итерация санындағы шешімді қайтарады.

Шектері бар СТЖ үшін функцияларды қолдануға арналған мысалдар.

2 мысал. Келесі теңдеулер жүйесін шешейік:

$$3x_1 + 4x_2 = 10$$
$$x_1 + 2x_2 = 15$$
$$7x_1 + 5x_2 = 3$$

```
» A = [3  4; 1  2; 7  5];
» B = [10; 15; 3];
» V = [0.4  0.1  0.5; 0.3  -0.2   0.5; 0.7   0.3   0.1];
» X = lscov(A, B, V)  →   X = [-3.3272   4.2531];
```

3 мысал. Келесі теңдеулер жүйесін шешейік:

3x+2y-5z=-1
2x-y+3z=13
X+2y-z=9

```
>> a=[3 2 -5; 2 -1 3; 1 2 -1]
a =
     3     2    -5
     2    -1     3
     1     2    -1
>> b=[-1;13;9]
b =
    -1
    13
     9
>> x1=lsqnonneg(a,b)
```

x1 =
   3.0000
   5.0000
   4.0000

4 мысал. Сиректелген матрицалары бар СТЖ-ні шешуге арналған функцияларды қолдану:

$$x_3 + 2x_4 = 11$$
$$x_1 + 3x_2 = 7$$
$$x_2 + x_4 = 6$$
$$x_1 + x_3 = 4$$

```
» A=[0  0  1  2; 1  3  0  0; 0  1  0  1; 1  0  1  0];
» B=[11; 7; 6; 4];
» lsqr(A, B, le-6.5)
lsqr converged at iteration 5 to a solution  with relative residual  1.9e-013
          ans = 1.0000   2.0000   3.0000   4.0000
```

5 мысал. *bicg(a,b)* функциясының көмегімен кедергілер градиенті екібағытты әдісін қолданып СТЖ-ні шешуге арналған мысал. Теңдеулер жүйесі берілген:

$$x_1 + 2x_2 + 3x_3 + 4x_4 = 4$$
$$5x_1 + 2x_2 + x_3 + 3x_4 = 6$$
$$4x_1 + 5x_2 + x_3 + 2x_4 = 8$$
$$x_1 + 9x_2 + 7x_3 + 3x_4 = 5$$

Жүйенің бастапқы матрицасын береміз:

```
» a=[1 2 3 4; 5 2 1 3 ; 4 5 1 2; 1 9 7 3];
» b=[4; 6; 8 ; 5];
» bicg(a,b)
  bicg converged at iteration 4 to a solution with relative residual 9.9e-015
 ans =    0.1278
          1.2026
         -1.4846
          1.4802
```

MATLAB-та СТЖ–ні шешуге арналған басқа да функциялар бар. Өзіндік жұмыстың тапсырмаларын орындау үшін:

1)берілген әдебиеттер тізімі бойынша СТЖ-ны шешуге арналған әдістерді менгеру;

2) сызықты теңдеулер жүйесінің көптеген түрін таңдап алуы қажет;

3) алынған сызықты теңдеулер жүйесін матрица түрінде жазу;

4) СТЖ-сін MATLAB-та шешу әдістерін үйрену;

5) MATLAB пакетінде бірнеше сызықты теңдеулер жүйесінің шешімін әртүрлі тәсілдерді қолданып , табу;

6) түсіндірме жазбаны толықтыру және өзіндік жұмысты қорғау.

1.2 Функция түбірлерін есептеу

f(x) = 0 немесе ***$f_1(x) = f_2(x)$*** түріндегі сызықты емес теңдеулерді шешу мәселесі жиі туындайды. Соңғы теңдеуді келесі түрге келтіруге болады:

$f(x) = f_1(x) - f_2(x) = 0$.

Осылайша, берілген тапсырма бір айнымалы *f(x)* функциясының *x* аргументінің мәнін табуға келтіріледі, бұл жағдайда функцияның мәні нөлге тең. Басқаша айтқанда мұндай тапсырманы функцияның түбірлерін табу есебі деп атайды.

MATLAB пакетінің бірқатар функциялары функциялармен жұмыс жасауға арналған[1-5]. Графикалық объектілердің дескрипторларымен ұқсастығы жағынан @ символының көмегімен берілетін функциялар дескрипторлары сыныптарының объектілері қолданыла алады, мысалы: fun=@exp. Функция деп құрамды функцияларды айтамыз, мысалы *sin(x)* немесе *exp(x)*, сонымен қатар қолданушының функциялары, мысалы m-файлдар-функциялар ретінде берілетін f(x).

Функцияның түбірлерін табуға арналған MATLAB-тың функциялары төменде келтірілген:

1) fzero(@fun, x) – *x* аргументінің бастапқы мәнінде символдық түрде берілген *fun* функциясы нөлге жеткендегі дәйекті *x* мәнін қайтарады; қайтарылған мән функция өзінің белгісін өзгерткендегі нүктеге жақын болады немесе мұндай нүктесі табылмаған жағдайда NaN-ға тең;

2) fzero(@fun,[xl x2]) - *fun(x)=0* болғандағы *x* мәнін қайтарады, іздеу интервалын бергенде *fun(x(1))* белгісі *fun(x(2))* белгісінен ерекшеленетіндей етіп*, x=[xl x2]* векторының көмегімен табамыз; егер бұл шарт орындалмаса қате шығарады; *fzero* функциясының итервалы бар болса, онда ол *fun* өзінің белгісін өзгертетін нүктеге жуық мәнді қайтарды;
3) fzero(@fun, x, tol) – tol берілген қателігіндегі нәтижені қайтарады.

fzero функциясы үшін нөл *fun* фунциясының графигі *x* өсін қиятын нүктесі болып саналады. **fzero** функциясының берілу түріне байланысты функцияның нөлін іздестіретін келесі жақсы танымал **сандық әдістер** іске асырылады: қиманы тең етіп бөлу әдісі, жанама әдісі және кері квадраттық интерполяция әдісі.

cos(x)=0 теңдеуін шешу мысалы:

```
» x = fzero(@cos, [1 3])
  x = 1.5708
```

fun1 функциясының түбірлерін іздеу мысалы. Қарастырылатын функция үшін *fun1.m* m-файлын құрамыз:

```
function f = funl(x)
f = 0.25*x + sin(x) - l;
```

Функцияның нөлдерін іздеу интервалын анықтау үшін осы функцияның графигін тұрғызайық (1 сурет):

» x = 0 : 0.1 : 10;
» plot(x, funl(x));
» grid on;

Суретте түбірлердің мәні [0.5   1], [2   3] және [5   6] интервалдарында берілгендігі көрсетілген. Оларды *fzero* функциясын қолданып, табайық:

» xl = fzero(@funl, [0.5   1])
  x1 = 0.8905
» x2 = fzero(@funl, [2   3])
  x2 = 2.8500
» x3 = fzero(@funl,[5 6])
  x3 = 5.8128
» x3 = fzero(@funl, 5, 0.001)
  x3 = 5.8111

1-сурет. *funl* функциясының графигі.

Өзіндік жұмыстың тапсырмаларын орындау үшін:

1)берілген әдебиеттер тізімі бойынша функцияның түбірлерін табуға арналған әдістерді менгеру;

2) түбірлерді табу үшін функцияның көптеген түрін таңдап алуы қажет;

3) функцияның түбірлерін іздестіруге арналған MATLAB-та шешу әдістерін үйрену;

4) MATLAB пакетінде қарастырылған функциялардың түбірлерін әртүрлі тәсілдерді қолданып табу;

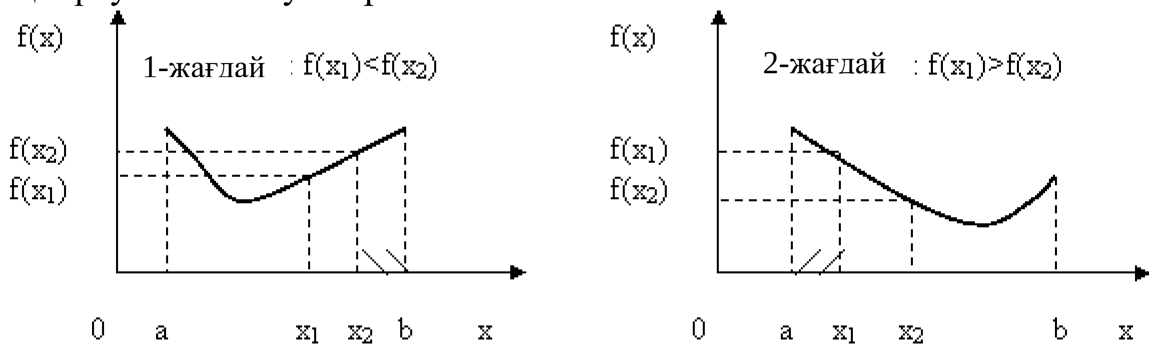
5) түсіндірме жазбаны толықтыру және өзіндік жұмысты қорғау.

1.3 Функцияларды минималдау

Сандық әдістердің маңызды тапсырмасы – кейбір интервалда x – x_1-ден x_2-ге дейін өзгергендегі *f(x)* функциясының минимумын іздеу болып табылады. Бұл жағдайда бірнеше әдістер қолданылады: алтын қима әдісі, көбейтіндіні қолданып минималдау әдісі, кіші квадраттар әдісі [1-5].

Алтын қима әдісінде қарастырылатын функция унимодалді болуы керек. Унимодалдіктің анықтамасын келтірейік: егер ***f(x)*** функциясының осы қимада жалғыз ғана глобалді минимум нүктесі болса және нүктенің сол жағында қатаң түрде кемімелі, ал оң жағында қатаң түрде өспелі ретпен нүктелер орналасса, онда ***f(x)*** функциясы ***[a, b]*** қимасында унимодалді болып табылады. Алтын қима әдісінің маңыздылығы мынада, ***[a, b]*** қимасында минималды қадамдар санында глобалді минимум нүктесін табу, яғни мақсаттық функцияны есептеудің минималды көлемі.

Берілген әдіске сәйкес уақыттың әрбір ағымдағы моменті үшін әрқашанда екі нүкте қарастырылады, мысалы, бастапқы моментте $a < x_1 < x_2 < b$ болатындай x_1 және x_2 нүктелері. Бұл жағдайда 2- суретте көрсетілген екі жағдайдың біреуі ғана болуы мүмкін.



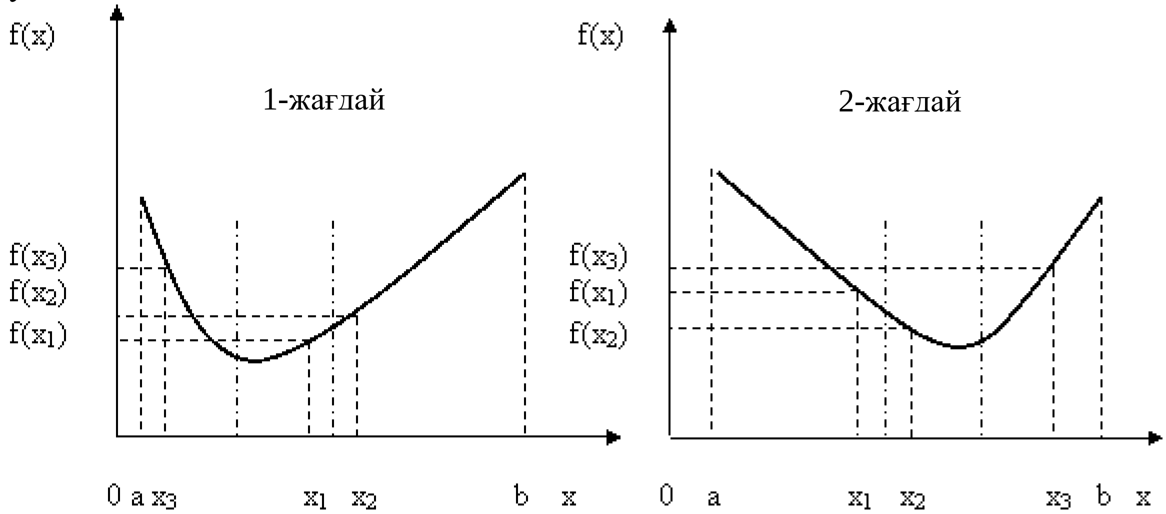
2- сурет. Қималарды шығарып тастау.

Бірінші жағдайда унимодалді функцияның қасиетіне байланысты x_{min} ізделінетін нүктесі $[x_2, b]$ қимасында жатуы мүмкін емес, ал екінші жағдайда $[a, x_1]$ (штрихталған) қимасында. Бұл дегеніміз, іздестіру облысы кішірейіп барады және келесі ізделінетін x_3 нүктесін қысқартылғын қималардың біреуінде: бірінші жағдайда - $[a, x_2]$ немесе екінші жағдайда - $[x_1, b]$.

Енді x_1 және x_2 нүктелерін бастапқы *[a, b]* қимасының қай жерінен таңдап алатынымызды анықтау қажет. x_{min} нүктесі туралы алғашқыда ешқандай мәлімет жоқ (графиктер жоқ және олар тұрғызылмайды, нақты жағдайда мақсаттық функцияның өрнегі ғана бар). Сондықтан да жоғарыда келтірілген оқиғалардың ықтималдылығы бірдей. Бұл дегеніміз, кез-келген қималардың біреуі артық: $[x_2, b]$немесе $[a, x_1]$. x_1 және x_2 нүктелерін *[a, b]* қимасының ортасына симметриялы түрде таңдап алу қажет.

Енді іздестіру облысын максималды түрде кішірейту үшін осы нүктелер бастапқы қиманың ортасына жақын орналасуы керек. Бірақ оларды ортасына өте жақын етіп те алуға болмайды, себебі біз мақсаттық функцияны есптеудің минималды санын алуды іске асыратын алгоритм тұрғызуымыз керек. 3-суретті қарастырайық.

Біріншіш қадамда салыстырылатын нүктелерді *[a, b]* қимасының ортасыны өте жақын етіп аламыз, сонда біз 1-жағдай үшін $[x_2, b]$ немесе 2- жағдай үшін $[a, x_1]$ үлкен қиманы қарастырудан шығарып тастаймыз. Бірақ екінші қадамда шығарылатын қиманың мөлшері шамалы ғана азаяды (1- жағдай үшін $[a, x_3]$ немесе 2-жағдай үшін $[x_3, b]$). Осылайша, алтын қима әдісінде бір жағынан нүктелерді қиманың ортасына жақын алу керек, екінші жағынан бір-біріне өте жақын алуға да болмайды.



3-сурет. Қимада нүктелердің орналасуы

Егер бір айнымалы функцияның максимумын табу керек болса, онда функцияның алдына «минус» таңбасын қою жеткілікті. Минимумды іздеу көп айнымалысы бар функцияларға да қолданылады.

Минимумды іздеу MATLAB-та *fminbnd* функциясының берілу түріне тәуелді танымал алтын қима әдісі немесе параболалық интерполяция әдісі арқылы орындалады [7-12]. Шешу үшін төмендегі функциялар қолданылады:

1) fminbnd (@fun,xl,x2) – *xl < x < x2* интервалындағы *fun(x)* функциясының локалді минимумы болатын *x* мәнін қайтарады;

2) fminbnd (@fun,xl,x2,options) – жоғарыда сипатталған функцияға ұқсас, бірақ *optimset* командасының көмегімен алдын-ала орнатылған *options* есептеу параметрлері векторының *tolX* (нақтылығы), *maxfuneval* , *maxiter* (итерацияның максималды саны), *display* параметрлерін қолданады.

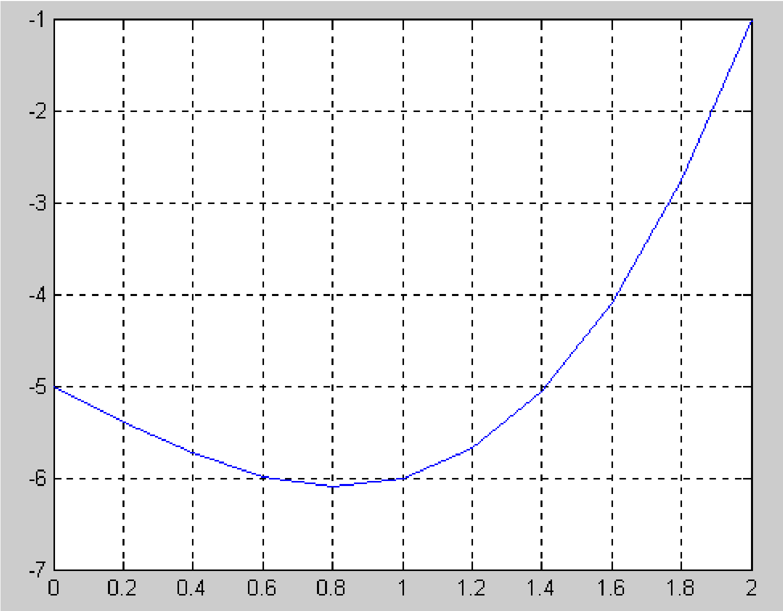
f(x) функциясы үшін минимумды іздеу мысалы. Функцияны m-файла түрінде жазайық:

```
function y = f(x)
y=x.^3 - 2*x - 5;
```

Минимумды анықтау үшін график тұрғызайық (4-сурет):

```
» x = 0 : 0.2 : 2;
» plot(x, f(x));
```

» grid on;
x мәнін және функцияның минимумын табайық:
» x = fminbnd(@fun, 0, 2)
   x = 0.8165
» y = f(x)
      y = - 6.0887



4-сурет. *f(x)* функциясының графигі.

Өзіндік жұмыстың тапсырмаларын орындау үшін:

1) берілген әдебиеттер тізімі бойынша функцияны минималдауға арналған әдістерді менгеру;
2) минимумын анықтауға арналған көптеген түрін таңдап алуы қажет;
3) функцияны минималдауға арналған MATLAB-та шешу әдістерін үйрену;
4) MATLAB пакетінде берілген әртүрлі тәсілдерді қолданып қарастырылған функциялардың графиктерін тұргызып, минимумдарын табу;
5) түсіндірме жазбаны толықтыру және өзіндік жұмысты қорғау.

1.4 Сандық интегралдау

$[a,b]$ қимасында $f(x)$ үздіксіз функциясының анықталған интегралын табу керек болсын:

$$I = \int_a^b f(x)dx$$

Сандық интегралдау есебін шешуге арналған бірнеше әдіс бар: тікбұрыш немесе трапеция формулаларының көмегімен, Симпсон формуласымен немесе

Гаусстың квадраттық формуласымен, сондай-ақ күрделі квадраттық формулалардың көмегімен есептейді [1-6].

Анықталған интегралды трапеция формуласының көмегімен есептеуді қарастырайық. $f \in C_2[0,h]$ болсын. Берілген

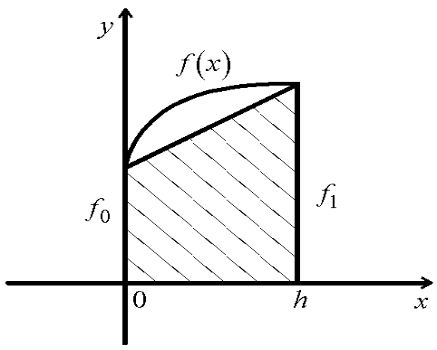
$$\int_0^h f(x)dx \approx h\frac{f_0 + f_1}{2},$$

мұндағы $f_0 = f(0), f_1 = f(h)$, яғни $\int_0^h f(x)dx$ интегралы 5-суретте көрсетілген штрихталған трапецияның ауданына жақындатылып өзгертіледі.

Трапецияның күрделенген квадраттық формуласының түрі:

$$\int_a^b f(x)dx \approx h(\frac{f_0}{2} + f_1 + f_2 + \ldots + f_{N-1} + \frac{f_N}{2}),$$

мұндағы, h - өзгерген қадамы, N - *[a, b]* интервалындағы нүтелер саны, f_i - i-нүтедегі функцияның мәні.



5-сурет. Қисық сызықты трапецияның ауданы

MATLAB пакетінің төменде келтірілген функциялары сандық интегралдауды трапеция әдісі және жинақтауы бар трапеция әдісін қолданып іске асырады:

1) trapz(Y) – есептеу барысында бір қадамдық болатын трапеция әдісін қолданып анықталған интегралды қайтарады. Егер *Y* – вектор болса, онда *trapz(Y)* – *Y* векторының элементтерінің интегралын қайтарады, егер *Y* – матрица болса, онда *trapz(Y)* осы матрицаның әрбір бағанының интегралынан тұратын вектор-жолды қайтарады;

2) trapz(X,Y) – трапеция әдісін қолданып *X* айнымалысы бойынша *Y* функциясының интегралын қайтарады (бұл жағдайда интегралдау шектері *X* векторының бастапқы және соңғы элементтерімен беріледі);

3) trapz(...,dim) – *dim* айнымалысының мәніне тәуелді бастапқы матрицасы үшін жолдары немесе бағандары бойынша интегралды қайтарады;

4) cumtrapz(Y) – инегралдау қадамы бірге тең (жинақтауы бар трапеция әдісімен интегралдау) Y матрицасы немесе векторының ординатымен берілген функция үшін анықталған интегралдың сандық мәнін қайтарады. Қадамы бірге

тең емес, бірақ тұрақты болған жағдайда есептелген интегралды қадамның шамасына көбейту жеткілікті. Вектор үшін бұл функция Y векторының элементттерін жинақтаған интегралдау нәтижесін вектор түрінде қайтарады. Матрицалар үшін Y матрицасының әрбір бағанының жинағы бар интегралдау нәтижесін беретін сол Y –тің өлшеміндегі матрицаны қайтарады;
5) cumtrapz(X, Y) – *Y*-тен *X* айнымалысына дейін трапеция әдісін қолданып интегралдауды орындайды. *X* және *Y* ұзындығы бірдей векторлар болуы керек немесе *X* вектор-баған, ал *Y* – матрица болуы керек;
6) cumtrapz(...,dim) *–dim* скалярымен нақты анықталған ұзындығына байланысты элементтер жинағы бар интегралдауды іске асырады. *X* векторының ұзындығы *size(Y, dim)* тең болу керек.

MATLAB-та сандық интегралдауды орындауға арналған басқа да функциялар бар.

trapz функциясын қолдану мысалы:

```
» trapz(cos(x), 2)
        ans =
                - 0.6410
                0.4369
                -   0.2241
```

quad функциясын қолдану мысалы:

```
» F = inline('1./(x.^3 - 2*x - 5)');
» Q = quad(F, 0, 2)
        Q =  - 0.4605
```

Өзіндік жұмыстың тапсырмаларын орындау үшін:

1) берілген әдебиеттер тізімі бойынша сандық интегралдауға арналған әдістерді менгеру;
2) сандық интегралдауға арналған көптеген функция түрін таңдап алуы қажет;
3) сандық интегралдауға арналған MATLAB-та шешу әдістерін үйрену;
4) MATLAB пакетінде берілген әртүрлі тәсілдерді қолданып қарастырылған функциялардың интегралдарының мәндерін табу;
5) түсіндірме жазбаны толықтыру және өзіндік жұмысты қорғау.

1.5 ҚДТ шешу

Қарапайым дифференциалды теңдеу (ҚДТ) деп $y=y(x)$ ізделетін функцияның бір немесе бірнеше алғашқы функциясы бар теңдеулерді атайды. Оларды келесі түрде жазуға болады:

$$F(x,y,y', \dots,y^{(n)})=0, \qquad (3)$$

мұндағы x – тәуелсіз айнымалы.

n көбейтіндінің жоғары ретін ҚДТ-дің реті деп атайды. Бірінші және екінші дәрежелі теңдеулерді жазайық:

$$F(x, y, y') = 0, \qquad F(x, y, y', y'') = 0.$$

(3) теңдеуінің шешімі $y=\varphi(x)$ кез келген функция болып табылады, оны теңдеуге қойсақ теңдік аламыз. Бірінші ретті теңдеу үшін жалпы шешім бір айнымалы тұрақтыға тәуелді болады:

$$y=\varphi(x, C).$$

Қарапайым дифференциалдық теңдеулерді шешу әдістерін келесі топтарға бөліп қарастыра аламыз: графиктік, аналитикалық, жуықталған және сандық. *Графиктік әдістер* геометриялық құрылуды пайдаланады, солардың бірі изоклин әдісі. Бірінші ретті ҚДТ қатары үшін *аналитикалық әдіс* шешімді аналитикалық түрлендіру жолымен алынған формулалар түрінде бере алады.

Жуықталған әдістер берілген теңдеуді қарпайым ету үшін кейбір мүшелерін негізді шығарып тастауды қолданады, сондай-ақ ізделінетін функцияның сыныптарын арнайы таңдау арқылы. Жуықталған әдістерде берілген тапсырмадағы кейбір кіші параметр арқылы шешімді қатарға қою пайдаланылады.

Кеңінен таралған және әмбебап сандық әдіс ретінде соңғы айырымдар әдісі саналады, оның маңыздылығы мынада. Аргументті үздіксіз өзгерту облысы (мысалы, қима) түйін деп аталатын нүтелердің дискреттік жиынымен ауыстырылады. Бұл түйіндер айырым торын береді. Ізделген үздіксіз аргументтің функциясы берілген тордағы дискретті аргументтің функциясы – тор функциясымен ауыстырылады. Бастапқы ҚДТ тор функциясына қатысты айырма теңдеуімен ауыстырылады. Мұндай ауыстыру торда ҚДТ-ны аппроксимациялау деп атайды. ҚДТ шешімі тордың түйіндерінде тор функцияларының мәнін іздеуге келтіріледі.

ҚДТ-ны шешу үшін Эйлер әдісін, Рунге-Кутта әдісін, Милн әдісін және көп қадамды әдісті қолдануға болады.

MATLAB-та ҚДТ жүйесін шешу үшін әртүрлі әдістер іске асырылған[7-11]. Олар ҚДТ шешулері деп аталған. Жалпы аты solver (шешуші), ҚДТ-ні шешудің сандық әдістерінің біреуін береді: ode45, ode23, ode113, ode15s, ode23s, ode23t , ode23tb, bvp4c немесе pdepe.

Шешушілер дифференциалды теңдеулер жүйесін шешудің келесі әдістерін іске асырады:

- ode45 – 4-ші және 5-ші ретті Рунге-Кутта бір қадамдық айқын әдісі; алғашқы шешуді сынауға ұсынылатын классикалық әдіс;
- ode23 – 2-ші және 4-ші ретті Рунге-Кутта бір қадамдық айқын әдісі; ҚДТ жүйесінің өте аз қатаңдығында және дәлдікке талап төмен болғанда бұл әдіс шешу уақыты жағынан ұтымды болады;
- ode113 – айнымалы ретті Адамса-Башворта-Мултонның көп қадамдық әдісі;шешімнің жоғары дәлдігін бере алатын адаптивті әдіс;
- ode23tb – шешу барысында Рунге-Кутта айқын емес әдісі және 2-ретті кері дифференциялдау формуласын қолданатын әдіс.

ҚДТ-ны шешу үшін басқа да функциялар бар.

1 мысал. Ван-Дер-Поль дифференциалды теңдеулер жүйесін шешу мысал. Теңдеулер жүйесі берілген:

$$y'_1 = y_2 \quad ,$$

$$y'_2 = 1000(1 - y_1^2)y_2 - y_1 \quad ,$$

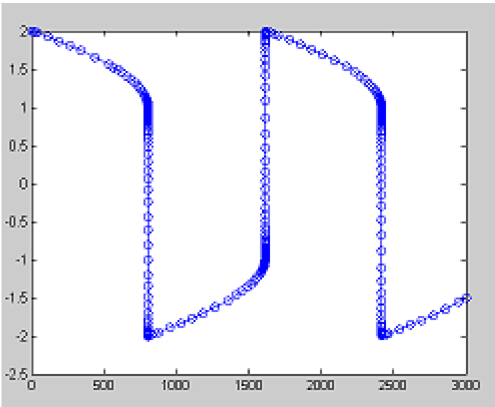
бастапқы шарттары $y_1(0)=0, \quad y_2(0)=1.$

m-файлды жазып аламыз:

```
function dy = vdp1000(t,y)
dy = zeros(2,1);
dy(1) = y(2);
dy(2) = 1000*(1 - y(1)^2)*y(2) - y(1);
```

MATLAB-та *ode15s* шешушінің көмегімен ҚДТ-ны шешу командасын жазып және осы функцияның графигін тұрғызамыз (6-сурет):

```
» [T,Y] = ode15s(@vdp1000,[0 3000],[2 0]);
» plot(T,Y(:,1),'-o')
```



6-сурет. Ван-Дер-Поль функциясының графигі.

2 мысал. Келесі теңдеулер жүйесін шешу мысалы:

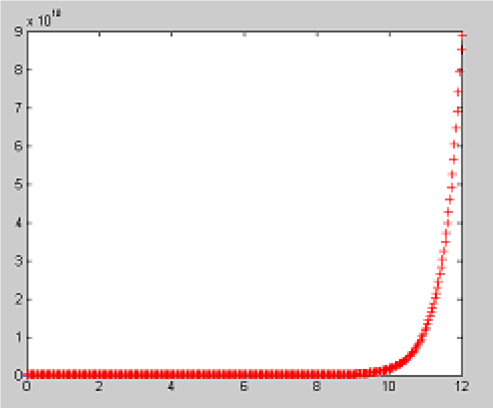
$$\frac{dx}{dt} = 2y; \qquad \frac{dy}{dt} = 2z; \qquad \frac{dz}{dt} = 2x;$$

m-файл құрамыз:

```
function f=rrr(t,y)
f=zeros(3,1);
f(1)=2*y(2);
f(2)=2*y(3);
f(3)=2*y(1);
```

MATLAB-та *ode23t* шешушінің көмегімен ҚДТ-ны шешу командасын жазып және осы функцияның графигін тұрғызамыз (7-сурет):

```
» options = odeset('RelTol', 1e-4, 'AbsTol', [1e-4 1e-4 1e-5]);
» [T,Y] = ode23t(@rrr, [0 12], [0 5 5], options);
» plot(T, Y, '+')
```



7-сурет. Функция графигі.

3 мысал. Дифференциалды теңдеуді шешу мысалы:

$$y' = 2x^2 + 2y\ ,$$

бастапқы шарттары $x_0 = 0$ болғанда $y_0 = 1$.

Бұл ҚДТ-ны шешетін командаларды MATLAB-та жазамыз:

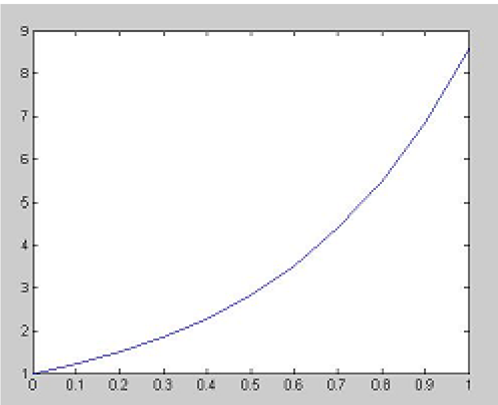
```
» y=dsolve('Dy=2*x^2 + 2*y', 'y(0)=1', 'x')
    y =  - x^2 - x - 1/2 + 3/2*exp(2*x)
```

Қарастырылған дифференциалды теңдеуінің алынған шешімі негізінде *du1* m-файлын жазайық:

```
I=1
for x = 0 : 0.1: 1
 u(I) = - x^2 - x - ½ + 3/2*exp(2*x);
 I = I+1;
   end
```

Алынған функцияның мәндерін есептеп және MATLAB-та графигін тұрғызайық (8-сурет):

```
» du1;
» x = [0 : 0.1: 1];
» plot(x,u)
```



8-сурет. Функция графигі.

Бақылау сұрақтары:

1. Қарапайым дифференциалдық теңдеулер қалай жазылады?
2. Қарапайым дифференциалдық теңдеулерді шешу үшін қандай әдістер пайдаланады?
3. MATLAB-та ҚДТ шешулері қалай жазылады?
4. MATLAB-та ҚДТ шешу үшін қандай пайдаланады?
5. MATLAB-та ҚДТ шешкенде бастапқы шарттар қандай түрде беріледі?

1.6 Полиномдармен жұмыс

x белгісізінің n-ші дәрежелі көпмүшелісі (немесе полином) деп келесі өрнек аталады:

$$a_0x^n + a_1x^{n-1} + \ldots + a_{n-1}x + a_n,$$

мұндағы $a_0, a_1, \ldots, a_{n-1}, a_n$ – айнымалы нақты немесе комплекс сандар болатын осы полиномның коэффициенттері, a_0 үлкен коэффициенті нөлден өзгеше болуы керек[1-6].

$f(x)$ және $g(x)$ екі көпмүшелі тең болып саналады (немесе тең теңдік):

$$f(x) = g(x)$$

егер белгісіздің бірдей дәрежелісінде коэффиценттері тең болса. Негізінде, жоқ дегенде бір коэффициенті нөлден жақсы болатын көпмүшенің ешқайсысы нөлге тең болуы мүмкін емес. Көпмүшелерді байланыстыратын теңдік белгісін осы көпмүшенің теңестірілген теңдеу ретінде қарастыру қажет. Осылайша n-ші дәрежелі көпмүшеліні өзінің коэффициенттерінің жиынымен анықталатын формальді өрнек ретінде қарастырамыз, $a_0, a_1, \ldots, a_{n-1}, a_n$, мұндағы $a_0 \neq 0$.

n-ші дәрежелі көпмүшелі кез келген n натурал сан үшін бар болады (нөлдік, квадраттық, кубтық және т.б. дәрежедегі көпмүшелі).

Егер ыңғайлы болу үшін *x*-ті өспелі дәреже ретімен жазылған екі $f(x)$ және $g(x)$ көпмүшелісі берілсе:

$$f(x) = a_0 + a_1x + \ldots + a_{n-1}x^{n-1} + a_nx^n, \quad a_n \neq 0,$$
$$g(x) = b_0 + b_1x + \ldots + b_{s-1}x^{s-1} + b_sx^s, \quad b_s \neq 0,$$

және егер, мысалы $n \geq s$ болса, онда олардың қосындысы деп келесі көпмүше айтылады:

$$f(x) + g(x) = c_0 + c_1x + \ldots + c_{n-1}x^{n-1} + c_nx^n,$$

олардың коэффициенттері $c_i = a_i + b_i$, $i=0, 1, \ldots, n$, бірдей белгісіз дәрежеден тұратын $f(x)$ *және* $g(x)$ көпмүшелерінің коэффициенттерінің қосындысымен алынады, егер $n>s$ болған жағдайда $b_{s+1}, b_{s+2}, \ldots, b_n$ коэффициенттерін нөлге тең деп аламыз. Егер s-тен n артық болса, қосындының дәрежесі n-ге тең болады, бірақ ол $n=s$ болғанда кездейсоқ n-нен кіші болуы да мүмкін, нақты айтқанда $b_n = - a_n$ жағдайында.

$f(x)$ және $g(x)$ көпмүшесінің көбейтіндісі деп төмендегі көпмүшені айтамыз:

$$f(x) * g(x) = d_0 + d_1x + \ldots + d_{n+s-1}x^{n+s-1} + d_{n+s}x^{n+s},$$

олардың коэффициенттері келесі түрде анықталады:

$$d_i = \sum_{k+i=i} a_k b_k, \quad i=0,\ 1,\ ...,\ n+s-1,\ n+s,$$

яғни, d_i коэффициенті - индекстерінің қосындысы i-ге тең $f(x)$ және $g(x)$ көпмүшелілердің коэффициенттерінің көбейтіндісінің нәтижесі және осыған ұқсас барлық көбейтінділердің қосындысы. Негізінде:

$$d_0 = a_0b_0,,\ \ d_1 = a_0b_1 + a_1b_0,,\ ...,\ d_{n+s} = a_nb_s.$$

Соңғы теңдіктен $d_{n+s} \neq 0$ теңсіздігі пайда болады, сондықтан да екі көпмүшенің көбейтіндісінің дәрежесі осы көпмүше дәрежелерінің қосындысына тең. Осылайша, нөлден өзгеше көпмүшелердің көбейтіндісі ешқашан нөлге тең болмайды.

Полиномдардың басқа да түрлері бар, мысалы, Тейлор көпмүшесі, Лагранждың интерполяциялық көпмүше және т.б.

Әдетте полиномды MATLAB-та төмендегідей жазады:

$$p(x) = a_nx\text{^}n + x_{n-1}x\text{^}n\ \text{-}1 + ... + a_2x\text{^}2 + a_1\text{^}x + a_0,$$

MATLAB-та полиноммен жұмыс жасауға арналған келесі функциялар бар:

- *poly(A)* –*n x n* өлшемді *A* матрицасы үшін элементтері *det(A-sI)* сипаттамалық полиномының коэфициенттері болатын, *n + 1* өлшемді вектор-жолды қайтарады, мұндағы *I* – бірлік матрица, ал *s* – Лаплас операторы;
- *poly(r)* – *r* векторы үшін түбірлері *r* векторының элементтері болатын полиномның коэффициенттерін беретін *r* вектор-жолды қайтарады; *roots(r)* – кері функция, оның нәтижесі бүтін санға көбейтілген *poly (r)*-ді береді.
- *roots (c)* – элементтері *c* полиномының түбірлері болатын вектор-жолды қайтарады.
- *polyval (p,x)* – массивте берілген нүктелерде есептелген *p* полиномының мәнін қайтарады; *p* полиномы – элементттері дәрежелерінің кему реті бойынша берілген полиномның коэффиценттерінен құралған вектор, *x* матрица немесе вектор болуы мүмкін; кезкелген жағдайда *polyval* функциясы *p* полиномының әрбір *x* элементінің мәнін есептейді.

Төмендегідей полиноммен жұмыс істеу мысалы:

$$x=7x^3 + 45x^2 + 12x + 23$$

MATLAB-та полиномды құрып және оның түбірлерін есептейік:

```
» x=[7, 45, 12, 23];
» d=roots(x)
   d =
     - 6.2382
     - 0.0952 + 0.7195i
     - 0.0952 - 0.7195i
```

полиномның коэффициенттерін қалпына келтіруге тырысамыз:

```
» B=Poly (d)
B =
```

1.0000   6.4286   1.7143   3.2857

полиномның коэффициенттері бүтін сан болу үшін, алынғын мәндерді 7-ге көбейтеміз:

» B*7

ans =

7.0000   45.0000   12.0000   23.0000

осы бастапқы полиномның коэффициенттері.

Бақылау сұрақтары:

1. Көпмүшелік қалай жазылады?
2. Қандай көпмүшеліктер тең болып саналады?
3. Көпмүшеліктердің қосындысы қалай есептеледі?
4. MATLAB-та көпмүшеліктер қалай өңделеді?
5. MATLAB-та көпмүшеліктің түбірлерін қандай команда көмегімен табуға болады?

1.7 Үш өлшемді графика галереясымен жұмыс істеу

Matlab пакетінің үш өлшемді графиканың галереясына кіру үшін Help менюінің командасын қолданамыз:

Examples and demos.

Үш өлшемді графика галереясы фигуралар және m – файлдармен көрсетілген, олардың тізімі 1-кестеде келтірілген. Әрбір фигуралар мен m – файлдарды оқып үйрену үшін келесі команданың көмегін пайдаланамыз:

type имя – m – файла

1-кесте

Matlab пакетінің үш өлшемді графика галереясының құрамы

Галереядағы аты	Файл	Фигураның аты
Knot	Knot.m	Байланған түйін
Quiver	Quivdemo. M	Векторлық көлемді өріс
Kleinl	Kleinl. M	Көлемді сақина
Cruller	Cruller.m	Мебиус көлемді сақинасы
Hoops	Tory4.m	Төрт көлемді құрсау
Slosh	Spharm2.m	Ұлуға ұқсас фигураны тұрғызу
Modes	Modes. M	Үшөлшемді беттің анимация фазаларын көрсету
Logo	Logo.m	MATLAB жүйесінің логотипін тұрғызу

Мысал ретінде «Байланған түйін» фигурасын қарастырайық, оны келесі команда бойынша шақырамыз:

kleinl

және командалық файлды ол үшін келесі команда көмегімен қарастырамыз:

type kleinl.m

Бір рет бұралған Мебиус көлемді лентаның графигін *kleinl* командасы тұрғызады. Бұл график біртүсті және жоғарыда оң жақта орналасқан жарықтық көзі арқылы фигураға жарық түсіруді имитациялайтын және жарық шағылысыуының эффектін іске асыратын фигураның функционалды бояуын көрсетеді.

Фигураның терезесінде негізгі менюдің командаларын пайдаланып фигураны барлық жағынан қарастыруға болады, сонан кейін командалық файлды өзгертіп фигураның қасиеттерін өзгертуге болады.

Осы фигураның m-файлының тексі:

```
ab = [0 2*pi]; %  фигураның параметрлері
rtr = [2 0.5 1];
pq = [40 40];
box = [-3 3 -3 3 -2 2];
vue = [55 60];
set(gcf,'color',[.7 .8 .9]);               %  фонның түсін береді
tube('xyklein',ab,rtr,pq,box,vue);         %   фигураны шығару
shading interp
colormap(pink);                            %   фигураның  түсін береді
 light
lighting phong
```

Өзіндік жұмыстың тапсырмаларын орындау үшін:

1) берілген әдебиеттер тізімі бойынша компьютерлік графиканың түрлерін және үш өлшемді графиканың мүмкіндіктерін менгеру;
2) MATLAB пакетінің үш өлшемді графика галереясына кіру және фигураларының құрамын зерттеу;
3) Фигуралардың барлық түрін зерттеу және олардың m – файлдарын да;
4) MATLAB пакетінде берілген әртүрлі тәсілдерді қолданып фигураларды өзгерту;
5) түсіндірме жазбаны толықтыру және өзіндік жұмысты қорғау.

2 СТУДЕНТТЕРДІҢ ӨЗІНДІК ЖҰМЫСТАРЫ БОЙЫНША ТАПСЫРМАЛАРДЫ ОРЫНДАУ

2.1 СӨЖ бойынша берілетін тапсырмалар

1-ші тақырып. MATLAB пакетінің анықтамалық жүйесімен жұмыс іс-теу.

Тапсырма. MATLAB пакетінің анықтамалық жүйесін оқып үйрену.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) **help** командасы арқылы MATLAB пакетіндегі интерактивті жәрдем жүйесін шақырып және папкалар тізбегін зерттеп, кейбіреулерін қарап шығу;
2) нақты объект және басты сөз арқылы анықтама алуды үйрену;
3) HTML файлы түрінде рәсімделген MATLAB пакетіндегі **HELP DESK** анықтамалық жүйесін зерттеу.

2-ші тақырып. Matlab пакетінің демонстрациялық мысалдармен жұмыс.

Тапсырма. Matlab пакетінің демонстрациялық мысалдармен жұмыс.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) *help demos* командасын қолдану арқылы MATLAB пакетіндегі демонстрациялық мысалдарын қарастырып, олардың кейбіреулерін іске қосу;
2) демонстрациялық мысалдарды орындалуын игеру.

3-ші тақырып. Matlab пакетінде векторларды өңдеу үшін қарапайым m-файлдарды құру.

Тапсырма. Matlab пакетінде векторларды өңдеу үшін m-файлдарды құру мүмкіндіктерін оқып білу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

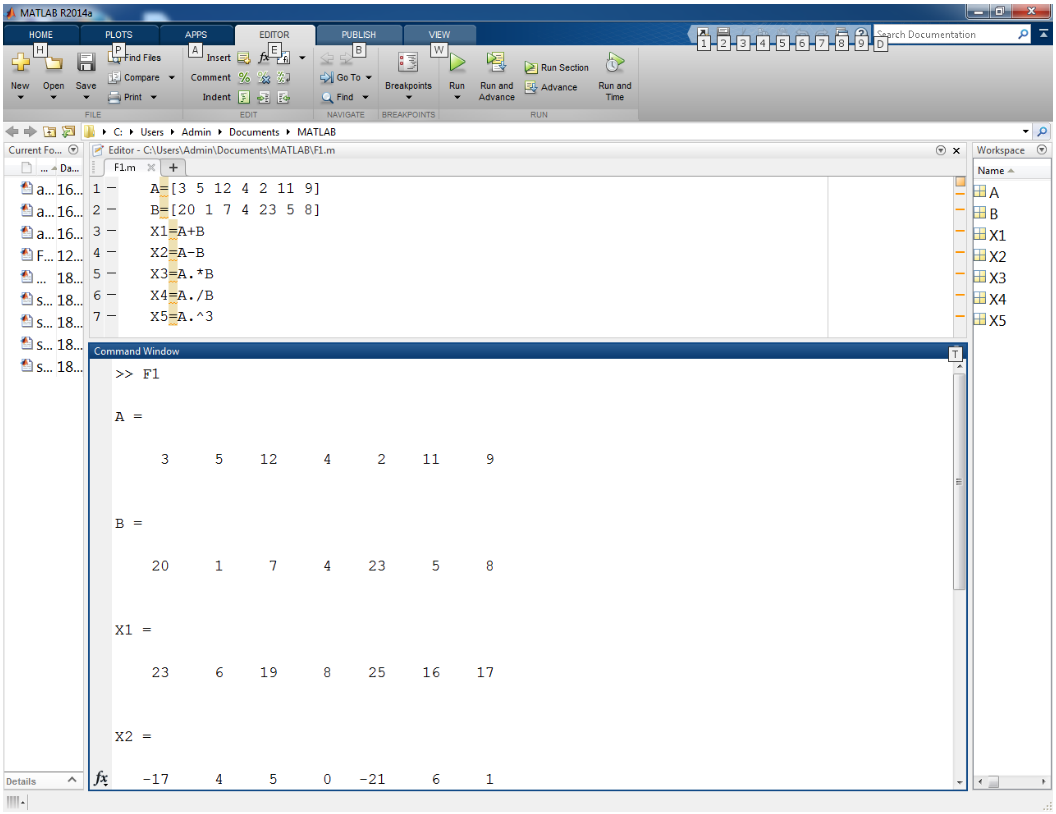
1) m-файлдарды құру режимін ашу үшін келесі команданы орындау керек:

   Файл/ Новый/ М-файл

2) ашылған терезеде *m-файлдың* мәтінін жазу керек, мысалы, векторларды өңдеу бойнша келесі командаларды жазу керек:

```
a=[1 2 3 4]
b=[4 5 6 7]
c=a.*b
t=0:pi/100:2*pi;
y=sin(t);
plot(t,y)
grid on
```

3) кез келген атымен осы файлды сақтау керек, мысалы, *aaa.m* (m-файлдың кеңейтілуі **.m**);
4) осы файлды келесі команда бойнша іске қосу керек: *Отладка/ Запуск;*
5) табылған қателерді түзету керек;
6) Matlab-тың командалар терезесіне қайта келіп, m-файлды келесі команда бойнша іске қосу керек: *aaa.m;*
7) Matlab-тың командалар терезесінде m-файлдың мәтінін келесі команда бойнша қарауға болады: type aaa.m;
8) векторларды өңдеу үшін бірнеше m-файлдарды құру керек(9 суретте m-файлдың мысалы келтіріл).



9-сурет. Векторларды өндеу бойынша m-файлдың мысалы

4-ші тақырып. Matlab пакетінде матрицаларды өңдеу үшін қарапайым m-файлдарды құру.

Тапсырма. Matlab пакетінде матрицаларды өңдеу үшін m-файлдарды құру мүмкіндіктерін оқып білу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) m-файлдарды құру режимін ашу үшін келесі команданы орындау керек: *Файл/ Новый/ М-файл*

2) ашылған терезеде m-файлдың мәтінін жазу керек, мысалы, матрицаларды өңдеу бойнша келесі командаларды жазу керек:

M=magic(4)
sum(M)
sum(diag(M))

3) кез келген атымен осы файлды сақтау керек, мысалы, a1.m ;

4) осы файлды келесі команда бойнша іске қосу керек: *Отладка/ Запуск*

5) табылған қателерді түзету керек

6) Matlab-тың командалар терезесіне қайта келіп, m-файлды келесі команда бойнша іске қосу керек: *a1.m*

7) Matlab-тың командалар терезесінде m-файлдың мәтінін келесі команда бойнша карауға болады: *type aaa.m*

8) матрицаларды өңдеу үшін бірнеше m-файлдарды құру керек.

5-ші тақырып. Векторлар үшін логикалық және математикалық стандартты функциялармен жұмыс.

Тапсырма. MATLAB пакетінде векторларды өңдеу бойынша логикалық және математикалық функциялардың мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетіндегі логикалық және математикалық стандартты функциялардың класстарын векторларды өңдеу үшін қолдану мүмкіндігін зерттеу;
2) зерттелген стандартты функцияларды көптеген мысалдарға қолданып, векторларды өңдеу үшін пайдалану керек.

6-ші тақырып. Векторлар үшін жиындарды өңдеу және жолдарды өндеу стандартты функциялармен жұмыс.

Тапсырма. MATLAB пакетінде векторларды өңдеу бойынша жиындарды өңдеу және жолдарды өндеу функциялардың мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетіндегі жиындарды өңдеу және жолдарды өндеу функциялардың векторларды өңдеу үшін қолдану мүмкіндігін зерттеу;

2) зерттелген жиындарды өңдеу және жолдарды өндеу функцияларды көптеген мысалдарға қолданып, векторларды өңдеу үшін пайдалану.

7-ші тақырып. Матрицаны өңдеуге арналған стандартты функциялармен жұмыс.

Тапсырма. MATLAB пакетінде матрицаны өңдеуге арналған стандартты функциялардың мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетіндегі стандартты функциялардың барлық класстарын матрицаны өңдеу үшін қолдану мүмкіндігін зерттеу;
2) зерттелген стандартты функцияларды матрицаны өңдеу үшін қолдану:
3) матрицаларды өңдегенде қандай функцияларды пайдалануға болмайды?

8-ші тақырып. Графиктерді редакциялау мысалдары: жазуларды және аңыздарды құру.

Тапсырма. MATLAB пакетінде екіөлшемді графиктерді редакциялағанда жазуларды және аңыздарды құру мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетіндегі екіөлшемді графиктерді редакциялағанда жазуларды және аңыздарды құру мүмкіндіктерін зерттеу;
2) зерттелген командаларды көптеген мысалдарға қолданып, екіөлшемді графиктерді редакциялағанда әртүрлі варианттарда пайдалану.

9-ші тақырып. Графиктерді редакциялау мысалдары: түсті және масштабты өзгерту.

Тапсырма. MATLAB пакетінде екіөлшемді графиктерді редакциялаганд түсті және масштабты өзгерту мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетіндегі екіөлшемді графиктерді редакциялағанда түсті және масштабты өзгерту мүмкіндіктерін зерттеу;

2) зерттелген командаларды көптеген мысалдарға қолданып, екіөлшемді графиктерді редакциялағанда әртүрлі варианттарда пайдалану.

10-ші тақырып. Үшөлшемді графиктің галереясындағы кез келген екі фигура үшін қасиеттерін өзгерту мысалдары.

Тапсырма. MATLAB пакетінде үшөлшемді графиктің галереясындағы кез келген екі фигура үшін қасиеттерін өзгерту мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетінің үшөлшемді графиктің галереясына кіру;

2) кез келген екі фигураларын түрлерін және оларға құрылған m-файлдарды зерттеу;

3) әрбір фигура үшін бірнеше қасиеттерін өзгерту керек, өзгерістерді қадам бойынша қарастырып.

11-ші тақырып. Статистикалық өңдеудің мысалдары: ковариация матрицасын және корреляция матрицасын құру.

Тапсырма. MATLAB пакетінде статистикалық өндеуде ковариация матрицасын және корреляция матрицасын құру мүмкіндіктерін зерттеу.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетінің мәліметтерді статистикалық өндеуде ковариация матрицасын және корреляция матрицасын құру командаларын игеру;

2)берілген матрица үшін ковариация матрицасын және корреляция матрицасын есептеу керек;

3) белгілі формула бойынша ковариация матрицасының мәліметтері бойынша корреляция матрицасын құруды тексеру керек (6 дәрісте келтірілген);

4)көптеген мысалдарда осындай есептеулерді орындау керек.

12-ші тақырып. Сандық интегралдауға арналған сандық әдістерді қолдану мысалдары.

Тапсырма. Сандық интегралдау үшін нақты сандық тәсілдерді қолдану бойынша MATLAB пакетінің мүмкіндіктерін оқып үйрену.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) MATLAB пакетіндегі сандық интегралдауға қолданылатын сандық тәсілдердің бірі болатын трапеция әдісін үйрену;

2) MATLAB пакетіндегі сандық интегралдауға қолданылатын сандық тәсілдердің бірі болатын жинақтку бар трапеция әдісін үйрену;

3) қарастырылып отырған әдістердің біреуімен қолдан есептеу жүргізу;

4) MATLAB пакетіндегі сандық интегралдаудың көптеген мысалдарда есептелуін орындау.

13-ші тақырып. Функцияларды аппроксимациялау үшін сандық әдістерді қолдану мысалдары: ең кіші квадраттар әдісі және бірқалыпты жуықтау әдісі.

Тапсырма. Функцияларды аппроксимациялау үшін нақты сандық әдістерді (ең кіші квадраттар және бірқалыпты жуықтау әдістерін) қолдану бойынша MATLAB пакетінің мүмкіндіктерін оқып үйрену.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) функцияны аппроксимациялау үшін сандық әдістер ретінде ең кіші квадраттар және бірқалыпты жуықтау әдістерін үйрену;
2) туындыларды шығару үшін қолданатын MATLAB пакетінің *diff* функциясын көптеген мысалдарда қолдану.

14-ші тақырып. Функцияларды аппроксимациялау үшін сандық әдістерді қолдану мысалдары: Ньютонның және Лагранждың көпмүшеліктерін.

Тапсырма. Функцияларды аппроксимациялау үшін нақты сандық әдістерді қолдану бойынша MATLAB пакетінің мүмкіндіктерін оқып үйрену: Ньютонның және Лагранждың көпмүшеліктерін.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) туындыларды аппроксимациялау үшін Ньютонның және Лагранждың интерполяциялық көпмүшелерінің қолдануын үйрену;
2) MATLAB пакетіндегі Лапласианның аппроксимация функцияларын зерттеп, оларды көптеген мысалдарда қолдану.
3) туындыларды шығару үшін қолданатын MATLAB пакетінің *diff* функциясын көптеген мысалдарда қолдану.

15-ші тақырып. Полиномдарды өңдеу үшін арналған сандық әдістерді қолдану мысалдары.

Тапсырма. Полиномдарды өңдеу үшін сандық әдістерді қолдану бойынша MATLAB пакетінің мүмкіндіктерін оқып үйрену.

Әдістемелік ұсыныс.

Тапсырманы орындау үшін келесі іс-әрекеттерді орындау керек:

1) полиномдардың берілу әдістерін үйрену;
2) полиномдармен жұмыс істегенде қолданылатын полиномдардың қосындысын есептеу;

көбейтіндіні табу, екі полиномдарды қалдықпен бөлу операцияларын үйрену;

3) полиномдарды өңдеу үшін қолданылатын негізгі операциялары бар MATLAB пакетінің функцияларын зерттеп, көптеген мысалдарда қолдану.

2.2 Жазбаша өзіндік жұмыстарды орындау

«Бизнес қосымшалардағы сандық әдістер» пәні бойынша жазбаша жұмыстарға 2 өзіндік жұмыстар жатады:

ӨЖ1- Вектор мен матрицаларды m-файлдар түрінде өңдеу;
ӨЖ2- MATLAB пакетінде сандық әдістер (СТЖ-ні шешу үшін, функцияларды минималдау, туындыларды аппроксимациялау, сандық интегралдау, полиномдармен жұмыс, қарапайым дифференциалды теңдеулерді шешу).

«Бизнес қосымшалардағы сандық әдістер» пәні бойынша өзіндік жұмыстарды орындау студентке математикалық есептеулерді жүргізу әдістерін қазіргі программалық пакеттерді компьютерде қолданудағы өзіндік білімін тексеруге және тереңдетуге көмектеседі. Сонымен қатар, жұмыс барысында студент бизнес қосымшалардағы және ақпараттық жүйелердегі сандық әдістерді іске асырудың принциптерін, статистикалық өңдеу іске асырудың және сандық әдістерді қолдану мүмкіндіктерін игереді, кәзіргі есептеу пакеттерді қолданып (MATLAB мысалында) және олардың көмегімен бизнес қосымшалардағы және ақпараттық жүйелердегі ғылыми-техникалық есептеулер үшін әртүрлі есептеулерді орындап, компьютерлік дайындық саласында өзінің кәсіптік сапаларын жақсартады.

ӨЖ1 орындауға дайындық мына кезеңдерден тұрады:
1) векторлар мен матрицаларды өңдеу бойынша MATLAB пакетінің мүмкіндіктерін оқып білу;
2) m-файлдарды құру режимін ашу, арифметикалық операциялар көмегімен векторларды өңдеу бойынша m-файлдарды құру, оны тексеріп іске қосу керек;
3) m-файлдарды құру режимін ашу, арифметикалық операциялар көмегімен матрицаларды өңдеу бойынша m-файлдарды құру, оны тексеріп іске қосу керек;
4) есептемені дайындау, өзіндік жұмысты қорғау.

ӨЖ2 орындауға дайындық мына кезеңдерден тұрады:
1) ӨЖ2 бойынша жеке тапсырма алу;
2) қойылған есепті орындау үшін теорияны және қолданылатын математикалық есептеулерді оқып білу;
3) ӨЖ2-де таңдалған жеке тапсырманы орындау бойынша MATLAB пакетінің мүмкіндіктерін оқып білу;
4) бастапқы мәліметтерді дайындау және MATLAB пакетінде бастапқы мәндер жиынында берілген есепке сәйкес есептеулер жүргізу;
5) барлық мысалдарды m-файлдар түрінде құру;
6) алынған нәтижелерді талдау;
7) есептемені дайындау, өзіндік жұмысты қорғау.

ӨЖ2 бойынша берілетін тапсырмалардың түрлері

1вариант. Сызықты теңдеулер жүйесін шешу. Төмендегілерді орындау қажет:

- сызықты теңдеулер жүйесін шешу әдістерін (Краммер әдісі, матрицалық әдіс, Гаусс әдісі және т.б.) үйрену және оларды сипаттау;
- үйренген әдістердің біреуін пайдаланып, қолмен есептеулер жүргізу;
- СТЖ шешуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

2 вариант. Функцияларды минималдау. Төмендегілерді орындау қажет:

- Функцияларды минималдау әдістерін (кесіндіні екіге бөлу әдісі, «алтын қиық» әдісі және т.б.) үйрену және оларды сипаттау;
- үйренген әдістердің біреуін пайдаланып, қолмен есептеулер жүргізу;
- функцияны минималдауға арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

3 вариант. Сандық интегралдау. Төмендегілерді орындау қажет:

- Сандық интегралдау әдістерін (тікбұрыштар формуласы, трапеция формуласы, Симпсон формуласы және т.б.) үйрену және оларды сипаттау;
- үйренген әдістердің біреуін пайдаланып, қолмен есептеулер жүргізу;
- сандық интегралдауға арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

4 вариант. Туындылардың аппроксимациясы. Төмендегілерді орындау қажет:

- туындыны аппроксимациялау әдістерін (Руни-Ромберг әдісі, анықталмаған коэффициенттер әдісі және т.б.) үйрену және оларды сипаттау;
- үйренген әдістердің біреуін пайдаланып, қолмен есептеулер жүргізу;
- туындыны аппроксимациялауға арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

5 вариант. Көпмүшелермен жұмыс. Төмендегілерді орындау қажет:

- көпмүшелерді жазу және оларды өңдеу әдістерін (Тейлор көпмүшесі, Лагранждың интерполяциялық көпмүшесі және т.б.) үйрену және оларды сипаттау;
- үйренген әдістердің біреуін пайдаланып, қолмен есептеулер жүргізу;
- көпмүшелермен жұмыс жасауға арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

6 вариант. Қарапайым дифференциялдық теңдеулерді шешу. Төмендегілерді орындау қажет:

- қарапайым дифференциялдық теңдеулерді шешу әдістерін (интегралдаудың жақын әдісі, Эйлер әдісін қолданып жақын шешу және т.б.) үйрену және оларды сипаттау;
- үйренген әдістердің біреуін пайдаланып, қолмен ҚТЖ есептеулер жүргізу;
- ҚТЖ шешуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

7 вариант. Деректерді интерполяциялау мен аппроксимациялау.

Төмендегілерді орындау қажет:

- деректерді интерполяциялау мен аппроксимациялау әдістерін үйрену және оларды сипаттау;
- деректерді интерполяциялау мен аппроксимациялауға арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

8 вариант. Сызықтық алгебраның матрицалық операциялары. Төмендегілерді орындау қажет:

- сызықтық алгебраның матрицалық операцияларын үйрену және оларды сипаттау;
- сызықтық алгебраның матрицалық операцияларына арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

9 вариант. Кесілген матрица функциялары. Төмендегілерді орындау қажет:

- кесілген матрица функцияларын жазу және өңдеу әдістерін үйрену және оларды сипаттау;
- кесілген матрица функцияларын жазуға және өңдеуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

10 вариант. Векторларға арналған операциялар. Төмендегілерді орындау қажет:

- векторларды жазу және өңдеу әдістерін үйрену және оларды сипаттау;
- векторларды жазу және өңдеу әдістерін жазуға және өңдеуге арналған -- MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

11 вариант. Матрицаларға арналған операциялар. Төмендегілерді орындау қажет:

- матрицаларды жазу және өңдеу әдістерін үйрену және оларды сипаттау;
- матрицаларды жазу және өңдеу әдістерін жазуға және өңдеуге арналған - MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

12 вариант. MATLABта графикалық объектілерді өңдеу қасиеттері. Төмендегілерді орындау қажет:

- компьютерлік графиканың түрлерін және оларды өңдеу әдістерін үйрену және оларды сипаттау;
- графиканы өңдеуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

13 вариант. MATLAB-тағы үшөлшемді графикалар галереясы. Төмендегілерді орындау қажет:

- компьютерлік графиканың түрлері мен оларды өңдеу әдістерін үйрену және оларды сипаттау;
- үшөлшемді графикалар галереясымен танысу;
- үшөлшемді графиканы өңдеуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

14 вариант. MATLAB-тағы арнайы графика. Төмендегілерді орындау қажет:

- компьютерлік графиканың түрлерін және оларды өңдеу әдістерін үйрену және оларды сипаттау;
- арнайы графикалармен танысу;
- арнайы графиканы өңдеуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың бірнеше функцияларын пайдаланып көптеген мысалдар келтіру.

15 вариант. MATLAB-та қолданушы интерфейсін жасау. Төмендегілерді орындау қажет:

- қазіргі таңдағы қолданушы интерфейсіне қойылатын талаптарды зерттеу және оларды сипаттау;
- қолданушы интерфейсін жасауға арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың әртүрлі мүмкіндіктерін пайдаланып қолданушының матрица, вектор, графикмен жұмыс дасауына арналған интрфейс жасау.

16 вариант. MATLAB-тағы екіөлшемді графиканы тұрғызу және оларды өңдеу. Төмендегілерді орындау қажет:

- компьютерлік графиканың түрлерін және оларды өңдеу әдістерін үйрену және оларды сипаттау;
- екіөлшемді графиканы өңдеуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың көптеген мүмкіндіктерін пайдаланып әртүрлі екі- өлшемді графикаларды тұрғызу және өңдеу мысалдарын келтіру.

17 вариант. MATLAB-тағы үшөлшемді графиканы тұрғызу және оларды өңдеу. Төмендегілерді орындау қажет:

- компьютерлік графиканың түрлері мен оларды өңдеу әдістерін үйрену және оларды сипаттау;
- үшөлшемді графиканы өңдеуге арналған MATLAB пакетінің мүмкіндіктерін зерттеу;
- MATLAB-тың көптеген мүмкіндіктерін пайдаланып әртүрлі үшөлшемді графикаларды тұрғызу және өңдеу мысалдарын келтіру.

Түсініктеме қағаз өзіндік жұмыс бойынша компьютерде басылған текст түріндегі және оларға қосымша қажетті түсініктемелер, суреттер, программа мәтіндері, бақылау мысалдарының басылымдарынан, блок-сұлбалардан тұратын барлық мағлұматтардан тұрады. ӨЖ1 және ӨЖ2 бойынша түсініктеме жазба мына бөлімдерден тұрады:

1) жеке тапсырма;
2) мазмұны;
3) кіріспе;
4) негізгі бөлім;
5) қорытынды;
6) әдебиеттер тізімі.

Кіріспеде жұмыс мақсаты, мәселенің актуалдығы, және сұрақтың зерттелу облысындағы қазіргі жағдайы жайлы жазу керек. ӨЖ2-ің негізгі бөліміне «ШЕШУДІҢ БАР ӘДІСТЕРІ» бөлімін енгізуге болады, мұнда теориялық түрде шешудің бар әдістерін, мысалға, (жеке тапсырмаға сәйкес) сызықтық теңдеулер жүйесін жазу керек және оған қосымша бір немесе бірнеше әдіс бойынша қолмен есептеу мысалын келтіру керек. ӨЖ2-на «MATLAB-тағы ШЕШУ ӘДІСТЕРІ» және «MATLAB-тағы ЕСЕПТЕУЛЕРДІ ОРЫНДАУ» бөлімдерін енгізуге болады, мұнда берілген есеп бойынша MATLAB мүмкіндіктерін сипаттау керек, бастапқы берілген мәндер жиынында жүргізілген MATLAB есептеулерінің нәтижесін жазу керек және осы нәтижелерді талдау керек. Қорытындыда алынған нәтижелер сапасына, MATLAB пакетінің тиімділігіне және оның әмбебаптығына баға беру қажет.

ӘДЕБИЕТТЕР ТІЗІМІ

1. Яблочкин Л.Б. и др. Основы численных методов. Уч. пос. – Тула: ТГУ, 2000.
2. Дьяконов В. MATLAB: учебный курс. – СПб.: Питер, 2001.
3. Самарский А. А. Численные методы: Уч. пособие для вузов. – М.: Наука, 1989.
4. Турчак Л.И. Основы численных методов: Уч. пос. – М.: Наука, 1987.
5. Ракитин В. И., Первушин В.Е. Практическое руководство по методам вычислений с приложением программ для персональных компьютеров. Уч. пос. – М.: Высшая школа, 1998.
6. Краскевич В.Е. и др. Численные методы в инженерных исследованиях. – Киев: Вища школа, 1986.
7. Потемкин В.Г. Система инженерных и научных расчетов MATLAB.Уч.-справ.изд. Т. 1 и 2 – М.: ДИАЛОГ-МИФИ, 1999.
8. Плис А.И., Сливина Н.А. Лабораторный практикум по высшей математике. Уч. пос. - М.: Высшая школа, 1983.
9. Абдуллина В.З. Компьютерные вычисления в пакете MATLAB. Методические указания к лабораторным работам и самостоятельной работе студентов по дисциплине «Компьютерные вычисления». – Алматы: КазНТУ, 2005.
10. Абдуллина В.З. Методы инженерных вычислений в MATLAB. Мет. указания к курс. проекту и самост. работе студентов по дис-нам «Компьютерные вычисления» и «Инженерные методы расчета» (для спец-тей 050703 и 050704). – Алматы: КазНТУ, 2006.
11. Мұртазина Ә.Ө., Абдуллина В.З. “MATLAB пакетіндегі инженерлік есептеулер әдістері» «Компьютерлік есептеулер» және «Есептеудің инженерлік әдістері» пәндері бойынша студенттердің курстық жоба менөзіндік жұмысына арналған әдістемелік нұсқауы.-Алматы:КазҰТУ, 2007.

МАЗМҰНЫ

2015 ж. жинақты жоспары, реті

Валентина Заманбековна Абдуллина
Әлия Өтебайқызы Мұртазина

ИНЖЕНЕРЛІК ЕСЕПТЕУЛЕРДІҢ САНДЫҚ ӘДІСТЕРІ

«Бизнес қосымшалардағы сандық әдістер» пәні бойынша студенттердің өзіндік жұмыстарына арналған әдістемелік нұсқаулар
5В070300 – «Ақпараттық жүйелер» мамандығының студенттері үшін

Редакторы

«Ақпараттық технологиялар» кафедрасы мәжілісінің хаттамасы № 8 « 7 » сәуір 2015 ж.

«АТТИ» институтының ОӘК мәжілісінің хаттамасы № 5 « 30 » сәуір 2015 ж.

Басуға __.__2015ж. қол қойылды.

Таралымы ___ дана. Пішіні 60х84 1/16. 1 типографиялық қағаз.
Көлемі __,__ ес. – б.т. Тапсырыс № __. Бағасы келісімді

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