6-amaliy mashg`ulot.

Mavzu: Matritsalarning asosiy xarakteristikalarini hisoblash. Reja

1. Amaliy mashg`ulot uchun kerakli jihozlar
2. Nazariy ma`lumotlar
3. MATLAB tizimining matrisaviy amallarga yo'naltirilganligi
4. Tizimlarni modellashtirish va loyihalashning turli bochqichlarida qo'llaniladigan modellar va modellashtirish vositalari.
5. Amaliy qism
6. Amaliy topshiriqlar

Kerakli jihozlar. Matlab®/Simulink®dasturiy ta'minoti bilan ta'minlangan kompyuterlar va printerlar.

MATLAB tizimining matrisaviy amallarga yo'naltirilganligi

MATLAB vaqt sinovidan o'tgan matematik hisoblarni avtomatlashtirish tizimlaridan biridir. U matrisaviy amallarni qo'llashga asoslangan. Bu narsa tizimning nomi- MATrix LABoratory-matrisaviy laboratoriyada o'z aksini topgan. Matritsalar MATLAB tizimining asosiy obyektlari hisoblanadi: 4.x versiyasida faqat bir o'lchamli massisvlar-vektorlar va ikki o'lchamli massivlar-matritsalar; 5.0 versiyasida ko'p o'lchhamli massivlardan-tenzorlardan foydalanish imkoniyati mavjud. MATLAB tizimi vektorlar va matrisalar ustida murakkab amallarni bajaradi. Undan arifmetik va algebraik amallardan tashqari matrisalarni invertirlash, ularning xususiy qiymatlarini hisoblash, chiziqli tenglamalar sistemasini yechish, ikki va uch o'lchamli funksiyalarning grafiklarini olish va boshqa ko'plab amallarni bajaruvchi kuchli kalkulyator sifatida ham foydalanish mumkin. Oddiy son va o'zgaruvchilarga ham MATABda 1x1 o'lchamli matrisa sifatida qaraladi. Shu sababli oddiy sonlar va massiv-lar ustida bajariladigan amallarning shakli va usullarida bir xillikka erishilgan. MATLAB - kengayuvchi tizim, uni har xil turdagi masalalarni yechishga oson moslashtirish mumkin. Uning eng katta afzalligi tabiiy yo'l bilan kengayishi va bu kengayish m-fayllar ko'rinishida amalga oshishidir. Boshqacha aytganda, tizimning kengayishlari kompyuterning qattiq diskida saqlanadi va MATLABning biriktirilgan (ichki) funksiyalari va proseduralari kabi kerakli vaqtda foydalanish uchun chaqiriladi. Zarur hollarda vektor va matrisalar massivlarga aylantiriladi va ularning qiymatlari har bir element uchun hisoblanadi. MATLAB oddiy arifmetik amallar va elementar funksiyalarni hisoblashdan tashqari vektorlar va matrisalar, kompleks sonlar, qatorlar va polinomlar bilan ham amallar bajaruvchi g'ayrioddiy superkalkulyatorga aylantiradi. Quyida massiv va matritsalar, matritaslar ustida

amallar, maxsus matritsalarning MATLAB tizimining 4.x versiyasida tashkil etish funksiyalari tavsifi keltirlgan:

Matritsalarni almashtirish amallari. Matlabda matritsalar ustida oddiy arifmyetik amallardan tashqari maxsus amallar va almashtirishlar mavjud. Ulardan biri matritsalarni transnponirlashdir. Biror A matritsani transponirlash dyeganda uni mos qatorlarini ustunlar bilan almashtirish tushuniladi va u A' kabi byelgilanadi. Masalan, A= [1 2 3; 4 5 6] bo'ladi. dyemak bunda (m*n) o'lchovli matritsaga o'tadi.

Bir nyechta matritsalarni birlashtirish uchun

V= cat (<o'lchov> A_1, A_2, ...)

komanda ishlatiladi. Bu holda A1, A2, ..., matritsalar ko'rsatilgan o'lchov bo'yicha birlashtiriladi:

cat (2, A, V) = [A, V] cat (1, A,
V) = [A; V]

Matlabda matritsalarni burish uchun fliplr (A), flipud (A) komandalaridan foydalaniladi. fliplr (A) komandasi A matritsani chapdan o'ngga ustunlarini almashtirish yo'nalishida buradi. flipud (A) esa A matritsani pastdan yuqoriga qatorlarini almashtirish yo'nalishida buradi. Masalan, A quyidagicha bo'lsin:

A= [2 3 7 1 9 0]

U holda fliplr (A) q [9 0; 7 1; 2 3] , flipud (A) q [3 2 ; 1 7; 0 9] kabi bo'ladi. Byerilgan matritsani soat stryelkasiga qarshi 900 ga buruvchi rot 90 (A) komandasidir.

Misol: B=[1 3 5 7 1 2 3 4];
rot 90(B)=[5 1 4 ; 3 9 3 ; 1
7 2];

Undan tashqari matlabda maxsus ko'rinishdagi matritsalarni hosil qilish imkoniyati bor. Ana shunday matritsalarni hosil qiluvchi komandalarni kyeltirib o'tamiz:

- size (A) – A matritsaning o'lchovi;
- length (A) –A vyektor uzunligi (elyemyentlar soni);
- ndims (A) – A matritsa o'lchovlari soni;
- isempty (A) – A matritsa bo'sh bo'lsa 1, aks holda 0 qiymatni byeradi;
- isegual (A, V) – A=V bo'lsa 1 ni byeradi, aks xolda "0" ni byeradi; inumeric (A) – A matritsa sonli tipda bo'lsa 1 ni byeradi, aks holda "0" ni byeradi; Namunalar:

1 – misol: Berilgan A va B matritsalarni bir biriga ko'paytirish:

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1]
          A =
-1    0     1
0   -1     0
1   -1     1
>> B=[1 1 0; 2 -1 0; 3 0 1]
          B =
-1    1     0
 2     -1         0
 3     0     1
>> A*B
ans =
     2    -1     1
    -2     1
0
     2     2
1
```

Endi shu amalni algoritmi haqida ya’ni o’z qo’limiz yordamida bajaramiz:

```
>> for i=1:3; for j=1:3; C(i,j)=0; for k=1:3; C(i,j)=C(i,j)+A(i,k)*B(k,j); end;
end; end; C C =
     2    -1     1
    -2     1     0
     2     2     1
```

2-Misol: A va B matritsalarni bir-biriga qo’shish

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1];
>> B=[1 1 0; 2 -1 0; 3 0 1];
>> A+B
ans =
     0     1     1
     2    -2     0
 4     -1     2
```

Endi shu matritsalarni qo’shish amalini algoritmini o’zimiz bajarib ko’ramiz: >> for i=1:3; for j=1:3; C(i,j)=A(i,j)+B(i,j);end; end; C

```
C =
     0     1     1
     2    -2     0
     4    -1     2
```

3 – misol: Matlabda matritsalarni chapdan o’ngga burishda fliplr komandasidan foydalanish:

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1]

A =
   -1     0     1

    0    -1
     0
    1     1
    -1     1
    >>
    fliplr(A
    ) ans =
2    0    -1

0    -1     0

1    -1     1
```

Endi shu komandani qo’lda bajarib chiqamiz:

```
>> for i=1:3; for j=1:3; C(i,j)=A(3-i+1,j);end; end; C
C =
1    -1     1
0       -
1     0 1
        0
1
```

4 - misol: Matlabda matritsalarni yuqoridan pastga burishda flipud komandasidan foydalanish:

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1]

A =
-1    0     1
0   -1     0
1   -1     1 >> flipud(A) ans =
    1    -1     1
    0    -1     0
 - 1     0     1
```

Endi shu amalni algoritmi bilan tanishib chiqamiz:

```
>> for i=1:3; for j=1:3; C(j,i)=A(j,3-i+1); end; end; C

C =
1     0    -1
0       -1     0
1       -1     1
```

5 – misol: Berilgan matritsani soat strelkasiga qarshi 90^0 ga burish uchun ishlatiladigan rot90(A) komandasi:

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1]

A =

-1     0     1
0    -1     0
1    -1     1 >> rot90(A) ans =
1     0     1
0    -1        -1
-1     0     1
```

Endi shu amalning bajarilish tartibi ya’ni algoritmi haqida:

```
>> for i=1:3; for j=1:3; C(i,j)=A(j,3-i+1); end; end; C

C =
1     0     1
0     -1         -1
1     0     1
```

Undan tashqari matlabda maxsus ko’rinishdagi matritsalarni hosil qilish imkoniyati bor. Ana shunday matritsalarni hosil qiluvchi komandalarni kyeltirib o’tamiz:

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1]

A =
-1     0     1
0    -1     0
1    -1     1 >> size(A) ans =
     3     3

>> length(A)
ans =      3
```

```
>>
ndims(A)
ans =
    2

>>
isempty(A)
ans =
    0
```

6 - misol: diag(A) komandasi berilgan matritsaning diagonalida tugan elementlarni ekranga chiqaradi:

```
>> A=[-1 0 1; 0 -1 0; 1 -1 1]
A =
   -1     0     1
      0        -1     0
      1        -1     1 >> diag(A) ans =
   -1      -1      1
```

Shu amalni o’zimiz bajarib chqamiz:

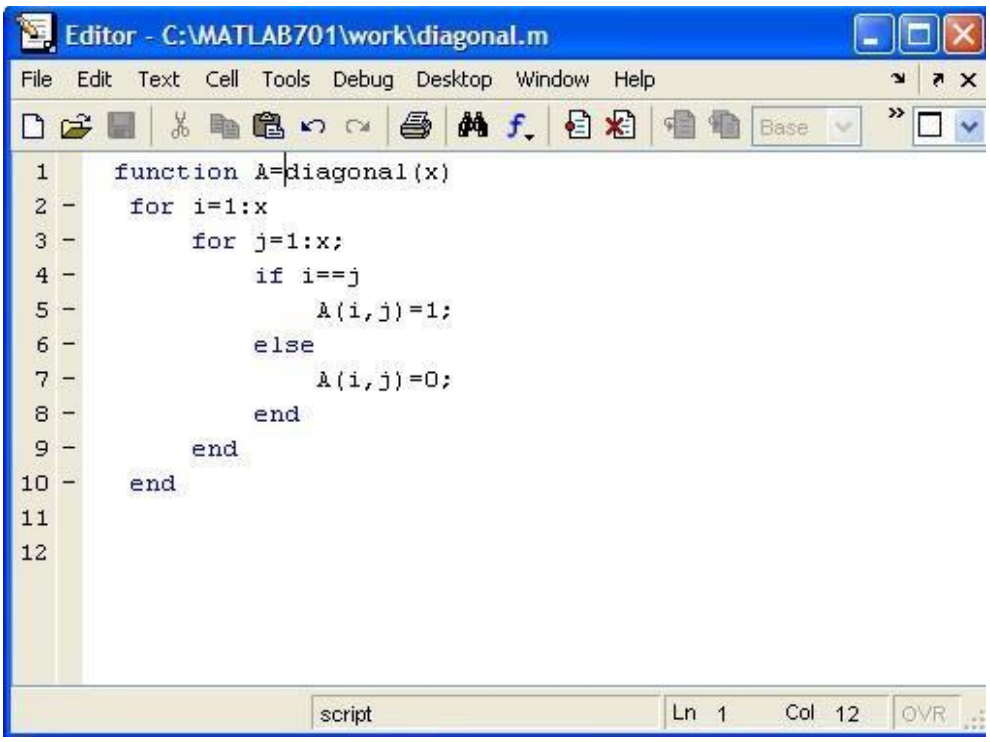
```
>> for i=1:3; D(i)=A(i,i);end; D

D =
   -1    -1     1
```

7 – misol: eye(n) komandasi birlik matritsa hosil qilish:

```
>> eye(5)
ans =
1     0     0     0     0
0     1     0     0     0
0     0     1     0     0
0     0     0     1     0
 0     0     0     0     1
```

Endi shu matritsani m-faylga funksiyasini yaratamiz:



Ushbu m-faylga birlik matritsa hosil qiladigan protsedura yasadik va uning nomini diagonal.m deb nomladik. Endi bu m-fayl yordamida diagonal(n) komandasi hosil bo'ldi. Endi ushbu komanda yordamida ham eye(n) komandasining bajargan ishini bajarsa bo'ladi: >> diagonal(5) ans =

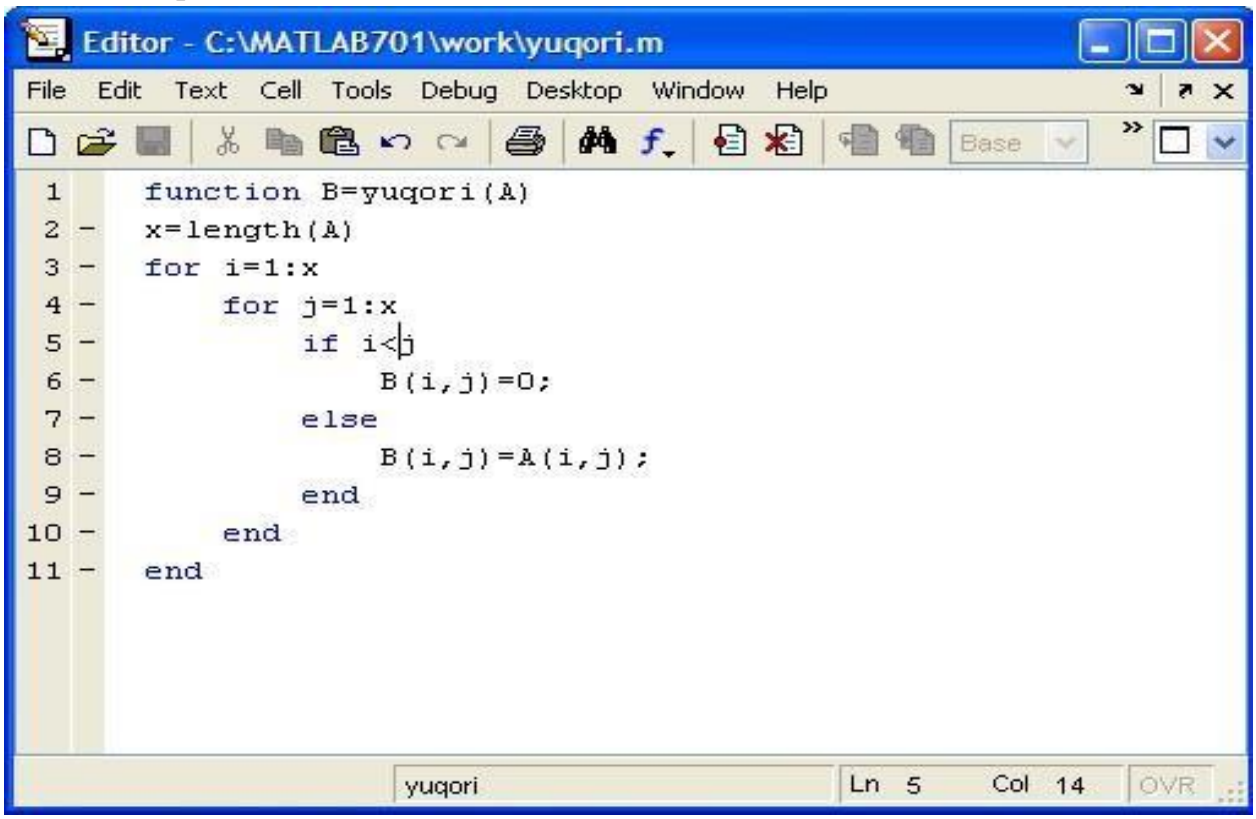
```
     0     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

8-misol: Berilgan matritsaning diagonaildan yuqori qismini elementlarini 0 bilan almashtirish:

```
>> A=[-1 2 4 0 3; -2 1 0 3 4; -2 -1 0 -2 1; -2 3 -1 -1 1; 1 1 1 -1 -1]
A =
    -1     2     4     0     3
    -2     1     0     3     4
    -2    -1     0    -2     1
    -2     3    -1    -1     1
     1     1     1    -1    -1
>>
tril(A)
ans =
```

-1 0 0 0 0
-2 1 0 0 0
2 -1 0 0 0
-2 3 -1 -1 0
1 1 1 -1 -1

Endi shu komandani o’zimiz m-faylga yozib yangi yuqori degan komanda hosil qilamiz :



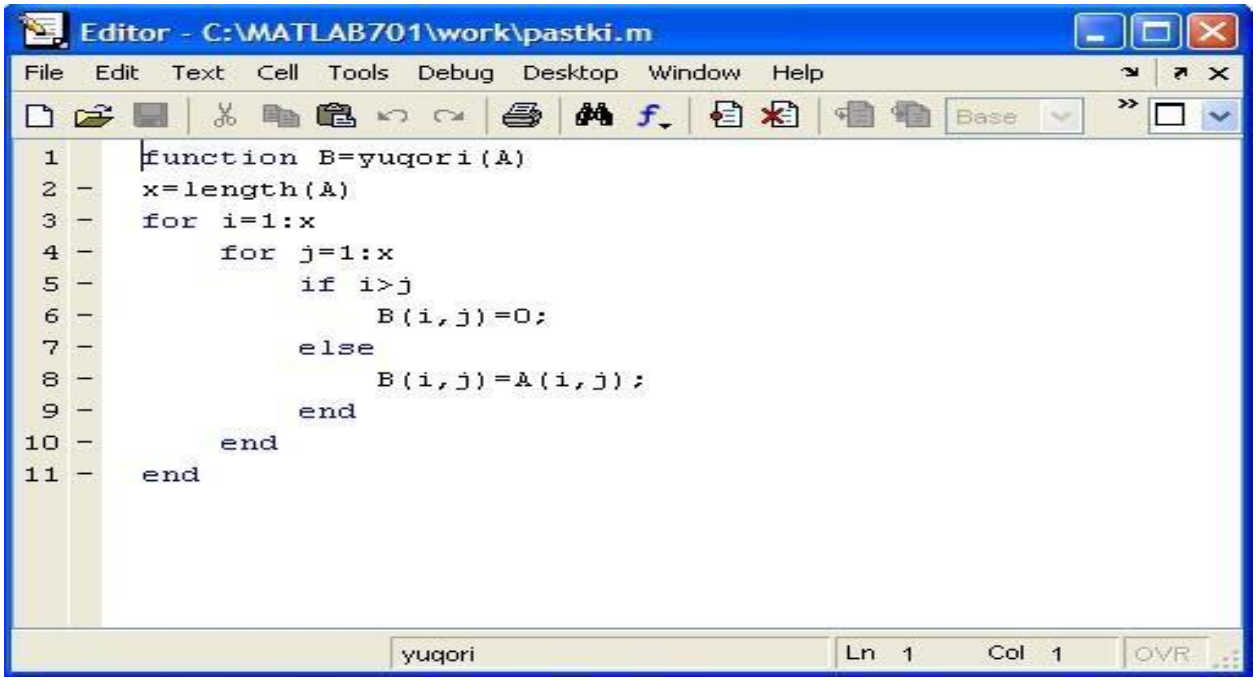
>> B=yuqori(A)
x =
5
B
=
-1 0 0 0 0
-2 1 0 0 0
-2 -1 0 0 0
-2 3 -1 -1 0
1 1 1 -1 -1

9 – misol :triu komandasi esa matritsaning diagonalidan pastki qismini nollarga aylantiradi:

>> A=[-1 2 4 0 3; -2 1 0 3 4; -2 -1 0 -2 1; -2 3 -1 -1 1; 1 1 1 -1 -1]

```
A =
-1    2    4    0
-2    1    0    3
 2   -1    0    174
 2   -1    1    174
 1   -1    1    175
>>
flip
ud(
A)
ans
=175
-1    2    4    0    3    178
 0    0    0   -1    1
 0    0    0    0   -1
```

Ushbu triu protsedurasini algoritmini o'zimiz tuzib m-faylga yozib chiqamiz va quyidagi natijalarga erishamiz:




```
>>
B=pastk
i(A) x =
     5

B =

-1    2    4    0    3
```

0 1 0 3 4

0 0 0 -2 1

0 0 0 -1

1 0 0 0 0

-1

10 – misol : **RESHAPE** – matrisa o'lchamini o'zgartish :

```
>> A=[-1 0 2 0; 0 1 2 -1; -1 -2 -3 2]
A =
   -1    0    2    0
    0    1    2   -1
    1   -2   -3    2 >> reshape(A,2,6)
ans =
   -1   -1    1    2   -3   -1
    0    0   -2    2    0    2
```

Talabalar mavzuni mukammal o'zlashtirishlari uchun bajaradigan topshiriqlar

1) **RESHAPE** – matrisa o'lchamini o'zgartiring : A=[-1 0 3 0; 0 1 2 -1; -4 -2 -3 2];
2) Berilgan matritsaning diagonaildan yuqori qismini elementlarini 0 bilan almashtiring:

A=[-1 2 4 0 3; -2 1 0 3 4; -2 -1 0 -2 1; -2 3 -1 -1 1; 1 1 1 -1 -1];

3) Berilgan matritsaning diagonaildan pastkii qismini elementlarini 0 bilan almashtirish:

A=[-6 2 4 4 3; -2 1 0 3 4; -2 -1 0 -2 1; -2 3 -1 -1 1; 1 8 1 -1 -1]

4) Berilgan matritsaning diagonaildan elementlarini 0 bilan almashtirish: A=[-2 2 4 0 3; -2 1 0 3 4; -2 -1 0 -2 1; -2 3 -1 -1 1; 1 5 1 -1 -1]
5) diag(A) komandasi berilgan matritsaning diagonalida tugan elementlarni ekranga chiqaring:

A=[-1 0 1; 0 -1 0; 1 -1 1]

6) diag(A) komandasi berilgan matritsaning diagonalida tugan elementlarni ekranga chiqaring:

A=[-3 5 1; 0 -1 0; 1 -6 1]

7) diag(A) komandasi berilgan matritsaning diagonalida tugan elementlarni ekranga chiqaring:

A=[-1 5 1; 0 -8 0; 1 -1 1]

8) Berilgan matritsani soat strelkasiga qarshi 90^0 ga burish uchun ishlatiladigan rot90(A) komandasi: A=[-1 0 1; 0 -1 0; 1 -1 1]
9) Berilgan matritsani soat strelkasiga qarshi 45^0 ga burish uchun ishlatiladigan rot45(A) komandasi: A=[-1 0 1; 0 -1 0; 1 -1 1]
10) Berilgan matritsani soat strelkasiga qarshi 180^0 ga burish uchun ishlatiladigan rot180(A) komandasi: A=[-1 0 1; 0 -1 0; 1 -1 1]