

3-amaliy mashg'ulot

Mavzu: Matrisalarning asosiy xarakteristikalarini

xisoblash. Reja

1. Amaliy mashg'ulot uchun kerakli jihozlar
2. Nazariy ma'lumotlar
3. Tadqiqot ob'yekti, ob'yekt matematik modeli, matematik modelni yechish algoritmi, algoritm asosida tuzilgan dastur.
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 massivlar-vektorlar va ikki o'lchamli massivlar-matritsalar; 5.0 versiyasida ko'p o'lchamli 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 MATLABda 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 oshirishidir. 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

matritsalarining MATLAB tizimining 4.x versiyasida tashkil etish funksiyalari tavsifi keltirilgan:

2. Matritsalar ustida amallar

Matritsalarini almashtirish amallari. Matlabda matritsalar ustida oddiy arifmetik amallardan tashqari maxsus amallar va almashtirishlar mavjud. Ulardan biri matritsalarini transponirlashdir. Biror A matritsani transponirlash dyeganda uni mos qatorlarini ustunlar bilan almashtirish tushuniladi va u A' kabi byelgilanadi.

Masalan, $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$ bo'ladi. dyemak bunda $(m \times n)$ o'lchovli matritsaga o'tadi.

Bir nyechta matritsalarini birlashtirish uchun

$V = \text{cat}(\langle \text{o'lchov} \rangle A_1, A_2, \dots)$

komanda ishlatiladi. Bu holda A_1, A_2, \dots , matritsalar ko'rsatilgan o'lchov bo'yicha

birlashtiriladi:

$\text{cat}(2, A, V) = [A, V]$ $\text{cat}(1, A, V) = [A; V]$

Matlabda matritsalarini 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:

$A = \begin{bmatrix} 2 & 3 & 7 & 1 & 9 & 0 \end{bmatrix}$

U holda `fliplr(A)` q $\begin{bmatrix} 9 & 0; & 7 & 1; & 2 & 3 \end{bmatrix}$, `flipud(A)` q $\begin{bmatrix} 3 & 2; & 1 & 7; & 0 & 9 \end{bmatrix}$ kabi bo'ladi. Byerilgan matritsani soat stryelkasiga qarshi 90 ga buruvchi `rot 90(A)` komandasidir.

Misol: $B = \begin{bmatrix} 1 & 3 & 5 & 7 & 1 & 9 & 2 & 3 \\ 4 \end{bmatrix}$; `rot 90(B)` $= \begin{bmatrix} 5 & 1 & 4; & 3 & 9 & 3; & 1 & 7 & 2 \end{bmatrix}$;

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

- `size(A)` – A matritsaning o'lchovi;
- `length(A)` – A vektor 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 matritsalarini 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 matritsalarini 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 matritsalarini 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 matritsalarini 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

```

```
>> 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 matritsalarini 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° 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 matritsalarini hosil qilish imkoniyati bor. Ana shunday matritsalarini 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  
>>  
ndim  
s(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
```

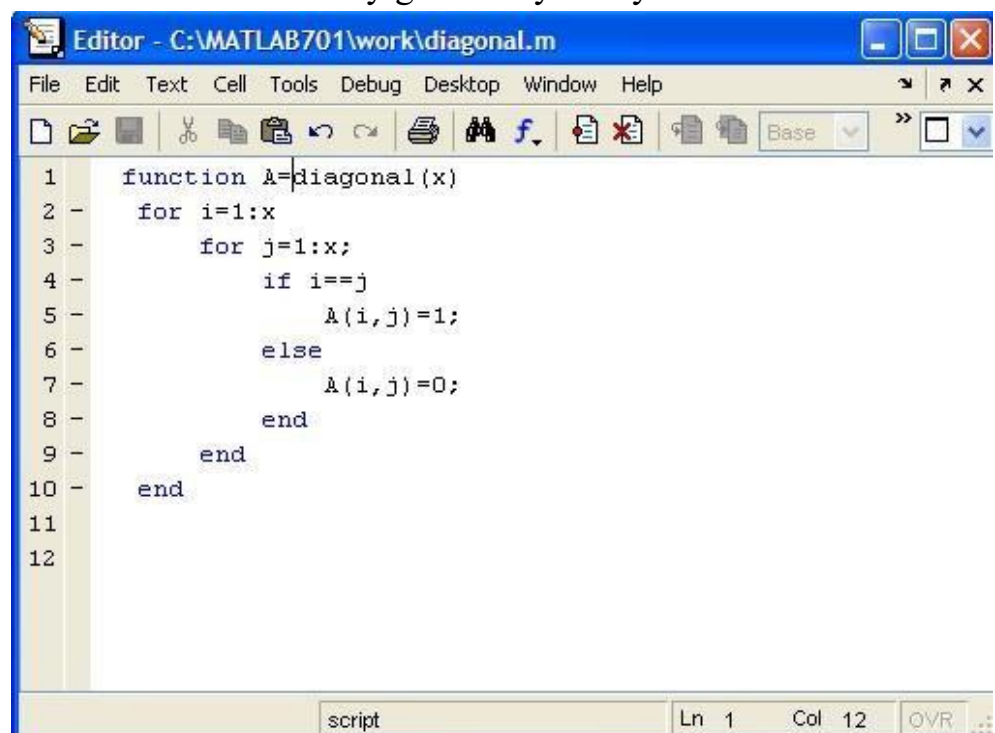
Shu amalni o'zimiz bajarib chqamiz:

```
>> 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     0     0
     1     0
     0     0     0     0     1
```

Endi shu matritsani m-faylga funksiyasini yaratamiz:



```
1 function A=diagonal(x)
2 -   for i=1:x
3 -       for j=1:x;
4 -           if i==j
5 -               A(i,j)=1;
6 -           else
7 -               A(i,j)=0;
8 -           end
9 -       end
10 -   end
```

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 =

```

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

```

8-misol: Berilgan matritsaning diagonaldan 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 =

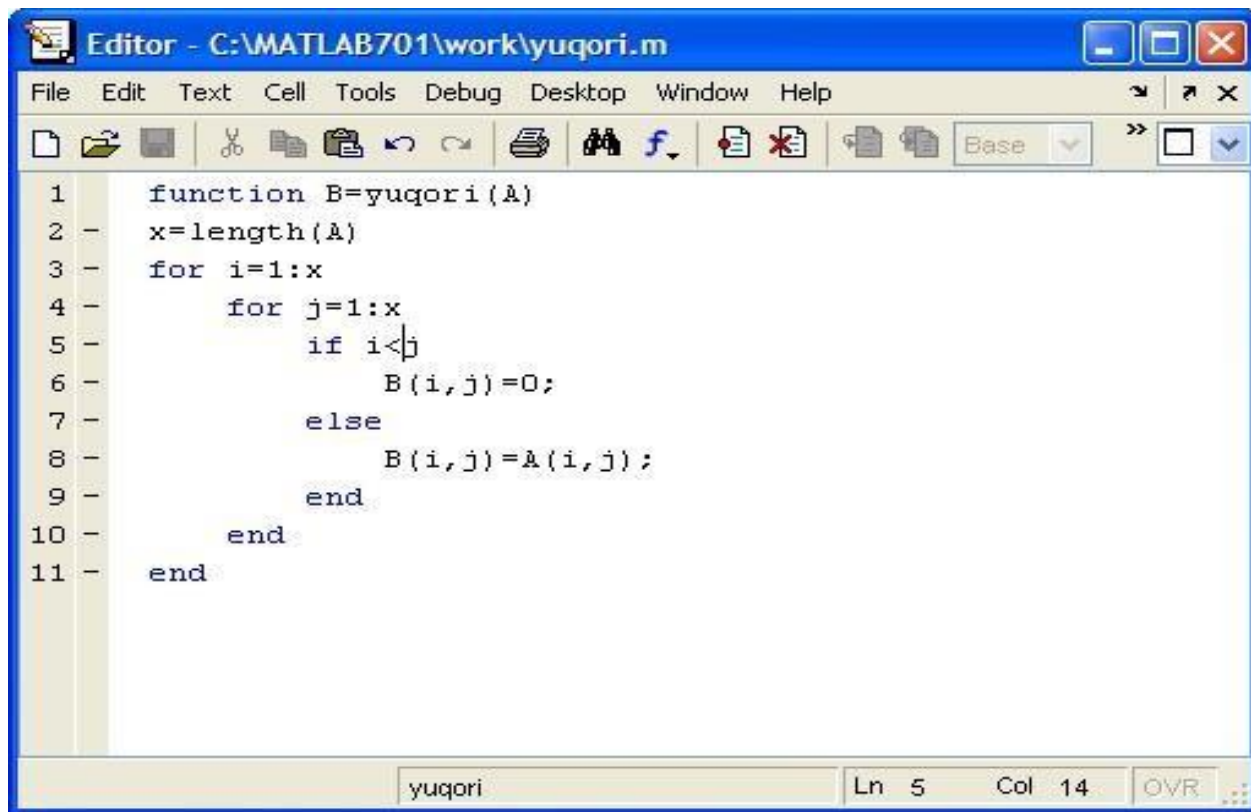
```

```

-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 :



```
Editor - C:\MATLAB701\work\yuqori.m
File Edit Text Cell Tools Debug Desktop Window Help
[Icons] Base >>
1 function B=yuqori(A)
2 x=length(A)
3 for i=1:x
4     for j=1:x
5         if i<j
6             B(i,j)=0;
7         else
8             B(i,j)=A(i,j);
9         end
10    end
11 end
yuqori Ln 5 Col 14 OVR
```

```
>> 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   147
 2   -1    1   147
 1   -1    1   148
```

```
>>
```

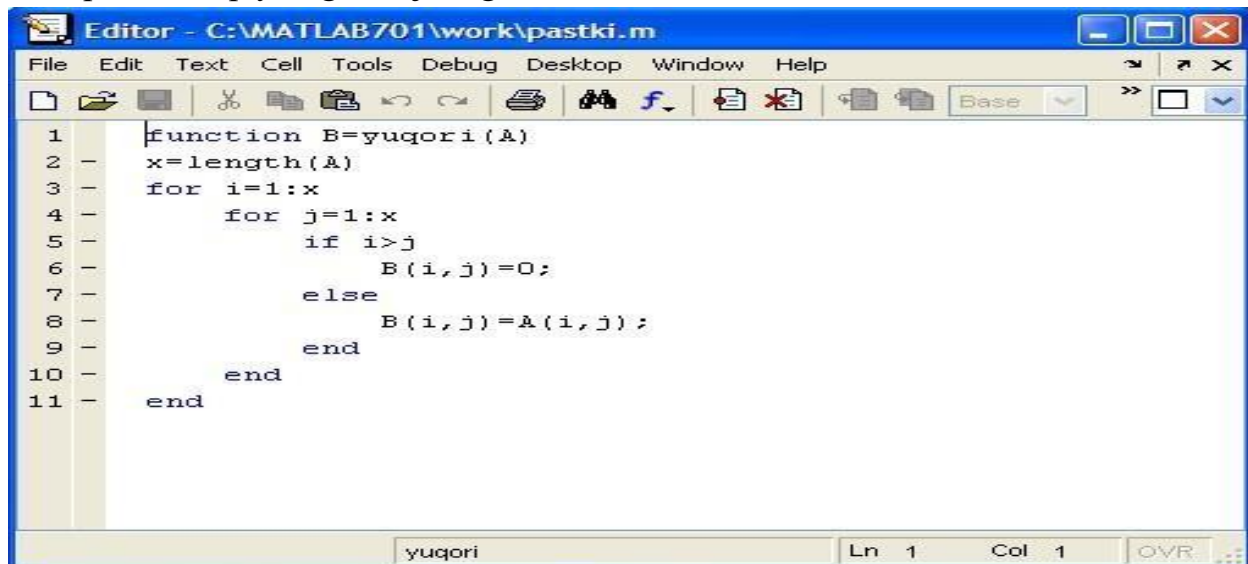


```

fl
i
p
u
d
(
A
)
a
n
s
= 148
-1  2  4  0  3 151
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:



The screenshot shows a MATLAB Editor window titled 'Editor - C:\MATLAB701\work\pastki.m'. The window contains the following MATLAB code:

```

1 function B=yuqori(A)
2 x=length(A)
3 for i=1:x
4     for j=1:x
5         if i>j
6             B(i,j)=0;
7         else
8             B(i,j)=A(i,j);
9         end
10    end
11 end

```

The status bar at the bottom indicates the file name 'yuqori', line number 'Ln 1', column number 'Col 1', and a status indicator 'OVR'.

```

>>
B=pas
tki(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

```

Ishni bajarish tartibi:

Laboratoriya mashg'ulotida har bir talaba ilovada keltirilgan masalalarni Matlab\Simulink dasturida yechishi va yechimlarini hisobot shaklida topshirishi talab etiladi.

Ilovadagi masalalar

- 1) **RESHAPE** – matrisa o'lchamini o'zgartiring : $A = [-1 \ 0 \ 3 \ 0; 0 \ 1 \ 2 \ -1; -4 \ -2 \ -3 \ 2]$;
- 2) Berilgan matritsaning diagonaldan 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 diagonaldan 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 diagonaldan 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:

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

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

$$A = \begin{bmatrix} -3 & 5 & 1 \\ 0 & -1 & 0 \\ 1 & -6 & 1 \end{bmatrix}$$

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

$$A = \begin{bmatrix} -1 & 5 & 1 \\ 0 & -8 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

8) Berilgan matritsani soat strelkasiga qarshi 90° ga burish uchun ishlatiladigan rot90(A) komandasi: $A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$

9) Berilgan matritsani soat strelkasiga qarshi 45° ga burish uchun ishlatiladigan rot45(A) komandasi: $A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$

10) Berilgan matritsani soat strelkasiga qarshi 180° ga burish uchun ishlatiladigan rot180(A) komandasi:

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

11) Matlabda matritsalarini yuqoridan pastga burishda **flipud** komandasidan foydalanish:

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

12) Matlabda matritsalarini chapdan o'ngga burishda flipr komandasidan foydalanish:

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

13) Matlabda matritsalarini o'ngdan chapga burishda flipr komandasidan foydalanish:

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix}$$

14) Matlabda matritsalarini chapdan o'ngga burishda flipr komandasidan foydalanish:

$$A = \begin{bmatrix} -7 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & -1 & 3 \end{bmatrix}$$

15) Matlabda matritsalarini chapdan o'ngga burishda flipr komandasidan foydalanish:

$$A = \begin{bmatrix} -3 & 0 & 3 \\ 0 & -7 & 0 \\ 1 & -2 & 1 \end{bmatrix}$$

16) A va B matritsalarini bir-biriga qo'shish

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 0 & -1 & 0 \\ 1 & -1 & 1 \end{bmatrix};$$

$$B = \begin{bmatrix} 1 & 1 & 0 \\ 2 & -1 & 0 \\ 3 & 8 & 1 \end{bmatrix};$$

17) A va B matritsalarini bir-biriga qo'shish

$$A = \begin{bmatrix} -1 & 7 & 1 \\ 0 & -5 & 0 \\ 1 & 2 & 1 \end{bmatrix};$$

$$B = \begin{bmatrix} 4 & 1 & 7 \\ 2 & -5 & 0 \\ 3 & 0 & 1 \end{bmatrix};$$

18) A va B matritsalarini bir-biriga ko'paytirish.

$$A = \begin{bmatrix} -1 & 7 & 1 \\ 0 & -5 & 0 \\ 1 & 2 & 1 \end{bmatrix};$$

$B = \begin{bmatrix} 4 & 1 & 7 \\ 2 & -5 & 0 \\ 3 & 0 & 1 \end{bmatrix};$

19) A va B matritsalarini bir-biriga bo'lish.

$A = \begin{bmatrix} -1 & 7 & 1 \\ 0 & -5 & 0 \\ 1 & 2 & 1 \end{bmatrix};$

$B = \begin{bmatrix} 4 & 1 & 7 \\ 2 & -5 & 0 \\ 3 & 0 & 1 \end{bmatrix};$

20) Matlab da for-loops va boshqa operatorlardan foydalanib n o'lchamdagi ixtiyoriy A matrisaning ustunlarining yig'indisini hisoblovchi **ustun_hisobi** deb nomlangan m-faylni (m-file) yarating.

Tekshirish uchun savollar:

- 1) Matematik modellash;
- 2) Matritsa va vektorlar;
- 3) Chiziqli algebra masalalari
- 4) Matritsa operatsiyalari va standart matritsalar;
- 5) Ikki va uch o'lchamli grafiklarni hosil qilish;
- 6) Dasturlash, m-fayllar va funksiyalar;
- 7) Shart operatorlari: if, else, elseif, switch, case.
- 8) Sikl operatorlari: for, while, continue, break;
- 9) Oddiy differensial tenglamalar;
- 10) Birinchi tartibli ODT, Eyler metodi;
- 11) Runge-Kutta metodi;
- 12) ODT yechilmalari: ode23, ode45, ode113;
- 13) Ikkinchi tartibli ODTlar va Yuqori tartibli ODTlar;
- 14) Prujina-massa-so'ndirgich tizimi;
- 15) Tabiiy tebranishlar va modal taxlillar;
- 16) Uzatish funksiyalari (Transfer Functions);
- 17) Vaqt domeni (Time Domain) va Chastota domeni (frequency Domain).