

Sistemi linearnih enačb

7. VAJA

7.1 Trikotni sistemi linearnih enačb

Izračunaj zgornje trikotni sistem linearnih enačb:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 0 & 1 & 1 \\ 0 & 0 & 2 \end{bmatrix}; \quad b = \begin{Bmatrix} 8 \\ 4 \\ 2 \end{Bmatrix};$$

in naredi preizkus.

Samostojno reši spodnje trikotni sistem:

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & -4.5 & 1 \end{bmatrix}; \quad b = \begin{Bmatrix} 7 \\ 13 \\ 5 \end{Bmatrix};$$

7.1 Trikotni sistemi linearnih enačb

```
Editor - C:\VAJENM~1\NUMERI~1\mm\VAJA9~1\vaja9_1a.m
File Edit Text Desktop Window Help
[Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons] [Icons]
1  %Primer reševanja zgornjega trikotnega sistema linearnih enačb
2  clc;clear all;
3  U=[ 1 2 3;0 1 1;0 0 2]
4  b=[ 8 4 2 ]'
5  n=length(b);
6
7  x(3)=b(3)/U(3,3);
8  x(2)=(b(2)-U(2,3)*x(3))/U(2,2);
9  x(1)=(b(1)-U(1,2)*x(2)-U(1,3)*x(3))/U(1,1);
10 disp ' ';disp 'Izpis rešitev'
11 fprintf('x(1) = %6.3f \r',x(1));
12 fprintf('x(2) = %6.3f \r',x(2));
13 fprintf('x(3) = %6.3f \r',x(3));
14
15 disp ' ';disp 'Splošen algoritem'
16 for i=n:-1:1
17     x(i)=(b(i)-dot(U(i,i+1:n),x(i+1:n)))/U(i,i);
18 end
19
20 fprintf('x(1) = %6.3f \r',x(1));
21 fprintf('x(2) = %6.3f \r',x(2));
22 fprintf('x(3) = %6.3f \r',x(3));
23
24 disp ' ';disp 'Preitkus'
25 p=U*x'
script Ln 1 Col 1 OVR
```

7.2 Gaussova eliminacija

Izračunaj sistem linearnih enačb z Gaussovo eliminacijo s pivotiranjem:

$$A = \begin{bmatrix} 1 & 4 & 1 \\ 1 & 6 & -1 \\ 2 & -1 & 2 \end{bmatrix}; \quad b = \begin{Bmatrix} 7 \\ 13 \\ 5 \end{Bmatrix};$$

in naredi preizkus.

7.2 Gaussova eliminacija

Gaussova eliminacija s pivotiranjem:

```
Editor - C:\WAJENM~1\NUMERI~1\nm\VAJA9~1\vaja9_2a.M
File Edit Text Desktop Window Help
1 %Reševanje sistema po Gaussovi m. s pivotiranjem
2 clc; clear all
3 A=[ 1 4 1; 1 6 -1; 2 -1 2];
4 C=A;
5 b=[ 7 13 5 ]'; A=[A,b]
6 %Pivotna vrstica je tista, ki ima v stolpcu največji element
7 disp 'Prvi korak'
8 [e,k]=max(A(1:3,1));
9 fprintf('Največji element v prvem stolpcu je %5.3f v %3i vrstici',e,k)
10 k=k+0;
11 if k~=1
12     pv=A(1,1:4);
13     A(1,1:4)=A(k,1:4);
14     A(k,1:4)=pv;
15 end
16 A
17 L(2,1)=A(2,1)/A(1,1); A(2,:)=A(2,:)-L(2,1)*A(1,:)
18 L(3,1)=A(3,1)/A(1,1); A(3,:)=A(3,:)-L(3,1)*A(1,:)
19 L
20 disp 'Drugi korak'
21 [e,k]=max(A(2:3,2));
22 k=k+1;
23 fprintf('Največji element v drugem stolpcu brez prve vrstice je %5.3f v %3i vrstici',e,k)
24 if k~=2
25     pv=A(2,2:4);
26     A(2,2:4)=A(k,2:4);
27     A(k,2:4)=pv;
28 end
script Ln 1 Col 1 OVR
```

```
Editor - C:\WAJENM~1\NUMERI~1\nm\VAJA9~1\vaja9_2...
File Edit Text Desktop Window Help
29 A
30 L(3,2)=A(3,2)/A(2,2)
31 A(3,2:4)=A(3,2:4)-L(3,2)*A(2,2:4)
32 %Rešitve
33 x(1:3)=0;
34 disp 'Računanje x(i)'
35 for i=3:-1:1
36     x(i)=(A(i,4)-dot(A(i,i+1:3),x(i+1:3)))/A(i,i)
37 end
38 disp 'Rešitev A\b'
39 A=C;x=(A\b)
script Ln 1 Col 1 OVR
```

7.2 Gaussova eliminacija

```
C:\vaje NM IR\numerične\mlvaja 9\vaja9_2b.m
File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + 1.1 x %>% %>% %>%
1 %Splošni algoritem
2 - clc; clear all
3 - A=[ 1 4 1; 1 6 -1; 2 -1 2];
4 - C=A;
5 - b=[ 7 13 5 ]'; A=[A,b]
6 - [m,n]=size(A);
7 - for i=1:m-1
8 -     for j=i+1:m
9 -         if(abs(A(i,i))<1.e-20)
10 -             error('matrika koeficientov je singularna')
11 -         end
12 -         [e,k]=max(A(i:m,i));
13 -         k=k+i-1;
14 -         if k~=i %zamenjava
15 -             pv=A(i,i:n); A(i,i:n)=A(k,i:n);
16 -             A(k,1:n)=pv;
17 -         end
18 -         L(j,i)=A(j,i)/A(i,i);
19 -         A(j,i)=0; A(j,i+1:n)=A(j,i+1:n)-L(j,i)*A(i,i+1:n);
20 -     end
21 - end
22 - x(1:m)=0;
23 - for i=m:-1:1
24 -     x(i)=(A(i,n)-dot(A(i,i+1:m),x(i+1:m)))/A(i,i);
25 - end
26 - disp(' ');disp('Rešitve')
27 - disp(x);
28 - disp('Rešitev A\b')
29 - A=C;x=(A\b)
script Ln 24 Col 50 OVR
```

Gaussova eliminacija
s pivotiranjem
(splošen algoritem):

7.3 Matlabove funkcije

$$A = \begin{bmatrix} 4 & -2 & -3 & 6 \\ -6 & 7 & 6.5 & -6 \\ 1 & 7.5 & 6.25 & 5.5 \\ -12 & 22 & 15.5 & -1 \end{bmatrix}; \quad b = \begin{Bmatrix} -12 \\ 22 \\ 15.5 \\ -1 \end{Bmatrix};$$

Reši zgornji sistem enačb z uporabo funkcij v Matlabu:

- operator \
- funkcija linsolve
- LU razcep
- QR razcep
- metoda konjugiranih gradientov
- uporaba inverzne matrike

7.3 Matlabove funkcije

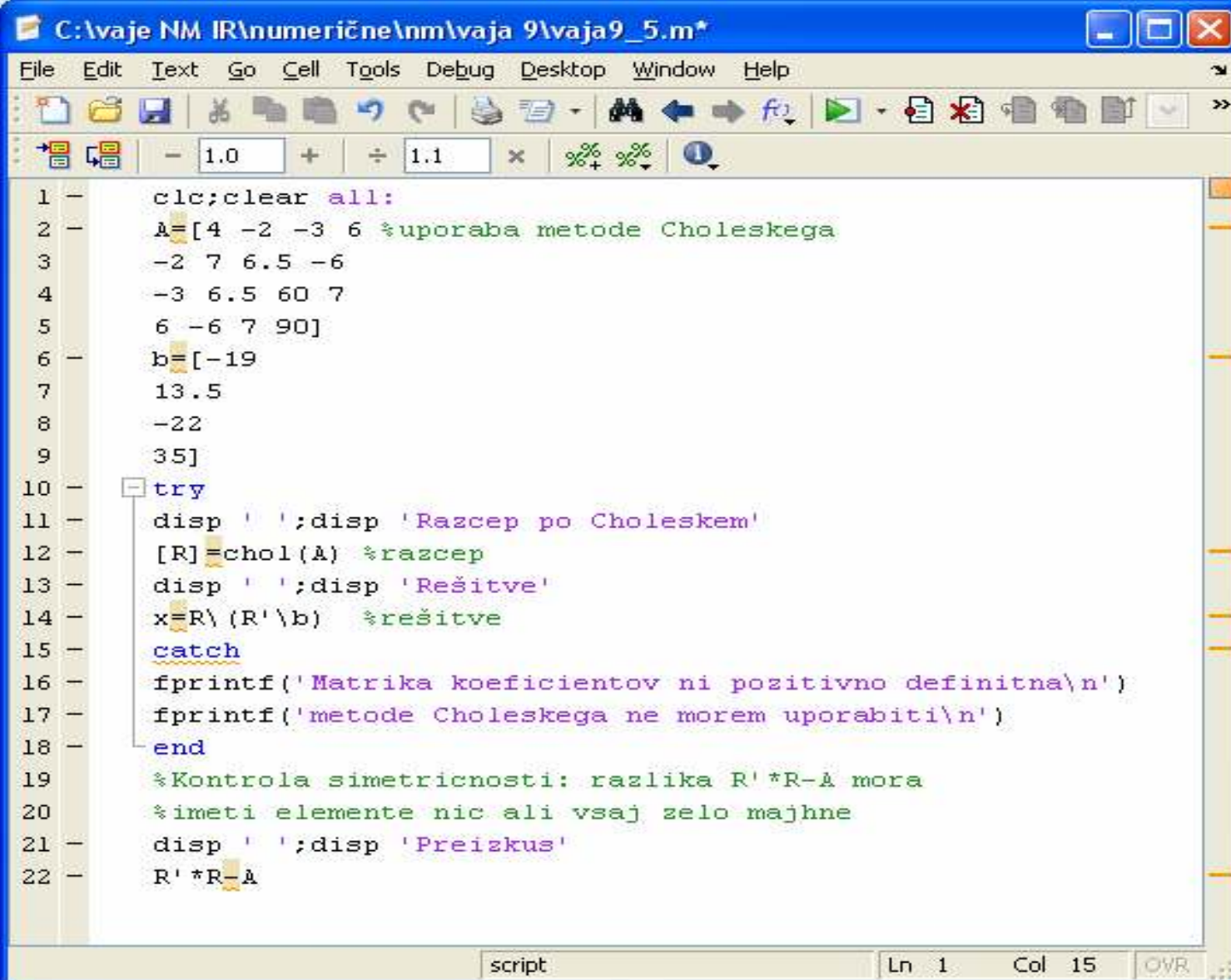
```
C:\vaje NM IR\numerične\m\l\vaja 9\vaja9_1b.m*
File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + ÷ 1.1 x %>% %>%
1 - clc;clear all;
2 - A=[4 -2 -3 6
3 - -6 7 6.5 -6
4 - 1 7.5 6.25 5.5
5 - -12 22 15.5 -1]
6 - b=[12 -6.5 16 17]
7 - b=b';
8 - disp ' ';disp 'Operator \'
9 - x=A\b %operator \
10 - disp ' ';disp 'Linsolve'
11 - x=linsolve(A,b) %funkcija linsolve
12 - disp ' ';disp 'LU razcep'
13 - [L,U,P]=lu(A) %lu razcep
14 - x=U\ (L\ (P*b))
15 - disp ' ';disp 'QR razcep'
16 - [Q,R]=qr(A) %qr razcep
17 - x=R\ (Q' *b)
18 - disp ' ';disp 'Rzširjen QR razcep'
19 - [Q,R,P]=qr(A) %razširjen qr razcep
20 - x=P*(R\ (Q' *b))
21 - disp ' ';disp 'Metoda konjugiranih gradientov'
22 - dop_nap=1.e-10 %metoda konjugiranih gradientov
23 - dop_st_it=25
24 - x=cgs(A,b,dop_nap,dop_st_it)
25 - r=A*x-b %preizkus
26 - disp ' ';disp 'Uporaba inverzne funkcije'
27 - x=inv(A)*b %uporaba funkcije inv
28
script Ln 1 Col 15 OVR
```


7.4 Metoda Choleskega

Reši pozitivno definiten in simetričen sistem linearnih enačb z uporabo metode Choleskega:

$$A = \begin{bmatrix} 4 & -2 & -3 & 6 \\ -2 & 7 & 6.5 & -6 \\ -3 & 6.5 & 60 & 7 \\ 6 & -6 & 7 & 90 \end{bmatrix}; \quad b = \begin{Bmatrix} -19 \\ 13.5 \\ -22 \\ 35 \end{Bmatrix};$$

7.4 Metoda Choleskega



```
C:\vaje NM IR\numerične\m\l\vaja 9\vaja9_5.m*
File Edit Text Go Cell Tools Debug Desktop Window Help
- 1.0 + ÷ 1.1 x % %
1 - clc;clear all:
2 - A=[4 -2 -3 6 %uporaba metode Choleskega
3 - -2 7 6.5 -6
4 - -3 6.5 60 7
5 - 6 -6 7 90]
6 - b=[-19
7 - 13.5
8 - -22
9 - 35]
10 - try
11 - disp ' ';disp 'Razcep po Choleskem'
12 - [R]=chol(A) %razcep
13 - disp ' ';disp 'Rešitve'
14 - x=R\(R'\b) %rešitve
15 - catch
16 - fprintf('Matrika koeficientov ni pozitivno definitna\n')
17 - fprintf('metode Choleskega ne morem uporabiti\n')
18 - end
19 - %Kontrola simetričnosti: razlika R'*R-A mora
20 - %imeti elemente nič ali vsaj zelo majhne
21 - disp ' ';disp 'Preizkus'
22 - R'*R-A
script Ln 1 Col 15 OVR
```