**Sprawozdanie z Laboratorium**

**z przedmiotu**

**Metody Numeryczne w Inżynierii**

**Ćwiczenie numer 2**

„Algebra liniowa”

**Informatyka**

Wtorek, tydzień x2, godzina 13:15 - 14:45

Grupa laboratoryjna: 1

Grupa dziekańska: 4I4

**Prowadzący: dr inż. Przemysław Mosiołek**

**Autorzy:**

Alan Buda – nr indeksu 211717  
Grzegorz Gronek – nr indeksu 202154

M-plik obliczający układ równań metodą Gaussa bez wyboru elementu głównego.

function [wynik] = gauss\_\_bez\_wyboru(A, b)

n = size(A,1);

A = [A b];

for k = 1:n-1

for i = k+1:n

A(i,:) = A(i,:)-(A(i,k)/A(k,k))\*A(k,:);

end

end

b = A(:,n+1);

A = A(:,1:n);

wynik=zeros(n,1);

for i = n:-1:1

wynik(i) = (b(i)-A(i,:)\*wynik)/A(i,i);

end

end

M-plik obliczający układ równań metodą Gaussa z częściowym wyborem elementu głównego.

function [wynik] = gauss\_\_z\_wyborem(A, b)

n = size(A,1);

A = [A b];

for k = 1:n-1

max=find(A(k:n,k)==wynik(A(k:n,k)))+k-1;

if(max > k )

tmp=A(k,:);

A(k,:)=A(max,:);

A(max,:)=tmp;

end

for i = k+1:n

A(i,:) = A(i,:)-(A(i,k)/A(k,k))\*A(k,:);

end

end

b = A(:,n+1);

A = A(:,1:n);

wynik=zeros(n,1);

for i = n:-1:1

wynik(i) = (b(i)-A(i,:)\*wynik)/A(i,i);

end

end

Obliczanie czasów operacji dla określonych rozmiarów macierzy.

function [] = program()

A2 = rand (2, 2)\*10;

b2 = rand (2, 1)\*10;

A4 = rand (4, 4)\*10;

b4 = rand (4, 1)\*10;

A8 = rand (8, 8)\*10;

b8 = rand (8, 1)\*10;

A16 = rand (16, 16)\*10;

b16 = rand (16, 1)\*10;

A32 = rand (32, 32)\*10;

b32 = rand (32, 1)\*10;

A64 = rand (64, 64)\*10;

b64 = rand (64, 1)\*10;

MNOZENIE\_A\_B = mnozenieAxb(A2, A4, A8, A16, A32, A64, b2, b4, b8, b16, b32, b64)

MNOZENIE\_A\_A = mnozenieAxA(A2, A4, A8, A16, A32, A64)

WYZNACZNIK\_A = wyznacznikA(A2, A4, A8, A16, A32, A64)

ODWROTNA\_A = odwrotnaA(A2, A4, A8, A16, A32, A64)

DZIELENIE\_A\_B = dzielenieAb(A2, A4, A8, A16, A32, A64, b2, b4, b8, b16, b32, b64)

TRACE\_A = traceA(A2, A4, A8, A16, A32, A64)

EIG\_A = eigA(A2, A4, A8, A16, A32, A64)\*0.2

wykresy(MNOZENIE\_A\_B, MNOZENIE\_A\_A, WYZNACZNIK\_A, ODWROTNA\_A, DZIELENIE\_A\_B, TRACE\_A, EIG\_A)

end  
  
A\*b

function [wynik] = mnozenieAxb(A2, A4, A8, A16, A32, A64, b2, b4, b8, b16, b32, b64)

wynik = zeros(1, 6);

tic

for i = 0 : 1000

A2\*b2;

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

A4\*b4;

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

A8\*b8;

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

A16\*b16;

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

A32\*b32;

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

A64\*b64;

end;

wynik(6)=toc/1000;

end

A\*A

function [wynik] = mnozenieAxA(A2, A4, A8, A16, A32, A64)

wynik = zeros(1, 6);

tic

for i = 0 : 1000

A2\*A2;

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

A4\*A4;

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

A8\*A8;

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

A16\*A16;

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

A32\*A32;

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

A64\*A64;

end;

wynik(6)=toc/1000;

end

det(A)

function [ wynik ] = wyznacznikA(A2, A4, A8, A16, A32, A64)

wynik = zeros(1, 6);

tic

for i = 0 : 1000

det(A2);

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

det(A4);

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

det(A8);

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

det(A16);

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

det(A32);

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

det(A64);

end;

wynik(6)=toc/1000;

end

**inv(A)**

function [ wynik ] = odwrotnaA(A2, A4, A8, A16, A32, A64)

wynik = zeros(1, 6);

tic

for i = 0 : 1000

inv(A2);

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

inv(A4);

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

inv(A8);

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

inv(A16);

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

inv(A32);

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

inv(A64);

end;

wynik(6)=toc/1000;

end

**A\b**

function [ wynik ] = dzielenieAb(A2, A4, A8, A16, A32, A64, b2, b4, b8, b16, b32, b64)

wynik = zeros(1, 6);

tic

for i = 0 : 1000

A2\b2;

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

A4\b4;

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

A8\b8;

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

A16\b16;

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

A32\b32;

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

A64\b64;

end;

wynik(6)=toc/1000;

end

**trace(A)**

function [ wynik ] = traceA(A2, A4, A8, A16, A32, A64)

wynik = zeros(1, 6);

x= [2 4 8 16 32 64];

tic

for i = 0 : 1000

trace(A2);

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

trace(A4);

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

trace(A8);

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

trace(A16);

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

trace(A32);

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

trace(A64);

end;

wynik(6)=toc/1000;

end

**eig(A)**

function [ wynik ] = eigA(A2, A4, A8, A16, A32, A64)

wynik = zeros(1, 6);

tic

for i = 0 : 1000

eig(A2);

end;

wynik(1) = toc/1000;

tic

for i = 0 : 1000

eig(A4);

end;

wynik(2)=toc/1000;

tic

for i = 0 : 1000

eig(A8);

end;

wynik(3)=toc/1000;

tic

for i = 0 : 1000

eig(A16);

end;

wynik(4)=toc/1000;

tic

for i = 0 : 1000

eig(A32);

end;

wynik(5)=toc/1000;

tic

for i = 0 : 1000

eig(A64);

end;

wynik(6)=toc/1000;

end

Wykresy:

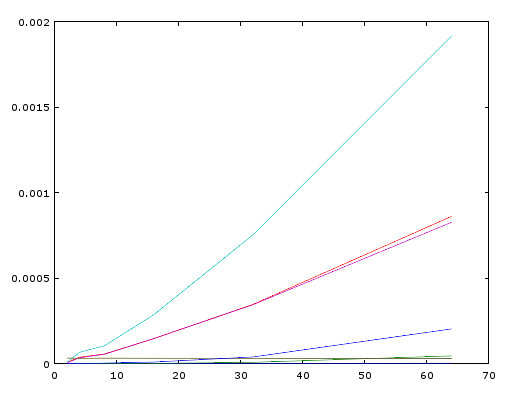
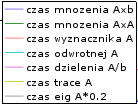
function [] = wykresy(wynik1, wynik2, wynik3, wynik4, wynik5, wynik6, wynik7 )

x = [2 4 8 16 32 64];

plot(x, wynik1, x, wynik2, x, wynik3, x, wynik4, x, wynik5, x, wynik6, x, wynik6, x, wynik7)

legend('czas mnozenia Axb', 'czas mnozenia AxA', 'czas wyznacznika A', 'czas odwrotnej A', 'czas dzielenia A/b', 'czas trace A', 'czas eig A\*0.2', 2)

end



**Wpływ błędów na rozwiązanie równania Ax = b**

**Wpływ zaburzeń na dokładność wyników w czterech przypadkach:**

1. Z wektora b zaburzony zostaje 1 element o 1%, 2%, 5%,10% i 20%
2. Z macierzy A zostaje zaburzony 1 element o 1%, 2%, 5%,10% i 20%
3. Cały wektor b zostaje zaburzony o 1%, 2%, 5%,10% i 20%
4. Jedna kolumna z macierzy A zostaje zaburzona o 1%, 2%, 5%,10% i 20%

function[] = program()

[A1, B1, X1] = tstmat(10);

[A2, B2, X2] = tstcnd(10);

zaburzenia(A1,B1,X1);

zaburzenia(A2,B2,X2);

end

function[] = zaburzenia(A,b,x)

**%zaburzenie jednego elementu wektora**

wynik1=zeros(1,5);

j = 1;

for i=1:20

if(i==1 || i == 2 || i == 5 || i == 10 || i == 20)

b1=b;

procent = i/100; b1(1)=b1(1)+(b1(1)\*procent);

x1=A\b1;

wynik1(j)=norm(x1-x)/norm(x); j = j +1;

end

end

figure(1)

plot(1:5, wynik1(1:5))

**%zaburzenie jednego elementu macierzy**

wynik2=zeros(1,5);

j = 1;

for i=1:20

if(i==1 || i == 2 || i == 5 || i == 10 || i == 20)

A1=A;

procent = i/100; A1(1,1)=A1(1,1)+(A1(1,1)\*procent);

x1=A1\b;

wynik2(j)=norm(x1-x)/norm(x);

j = j +1;

end

end

figure(2)

plot(1:5,wynik2(1:5))

**%zaburzenie calej kolumny macierzy**

wynik3=zeros(1,5);

j = 1;

for i=1:20

if(i==1 || i == 2 || i == 5 || i == 10 || i == 20)

A1=A;

p=i/100;

A1(:,1)=A1(:,1)+p\*norm(A1(:,1))\*rand(10,1);

x1=A1\b;

wynik3(j)=norm(x1-x)/norm(x); j = j +1;

end

end

figure(3)

plot(1:5,wynik3(1:5))

**%zaburzenie calego wektora**

wynik4=zeros(1,5);

j = 1;

for i=1:20

if(i==1 || i == 2 || i == 5 || i == 10 || i == 20)

p=i/100;

b1=b+p\*rand(10,1)\*norm(b);

x1=A\b1;

wynik4(j)=norm(x1-x)/norm(x);

j = j + 1;

end

end

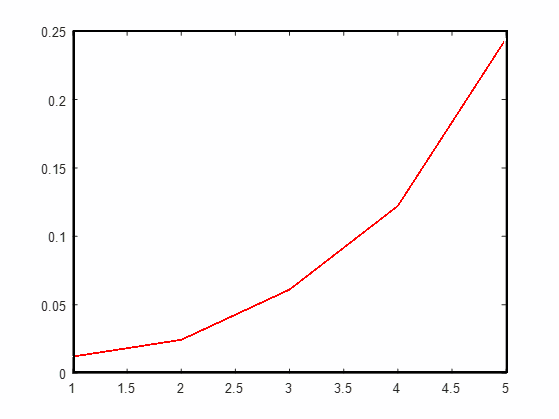
figure(4)

plot(1:5, wynik4(1:5))

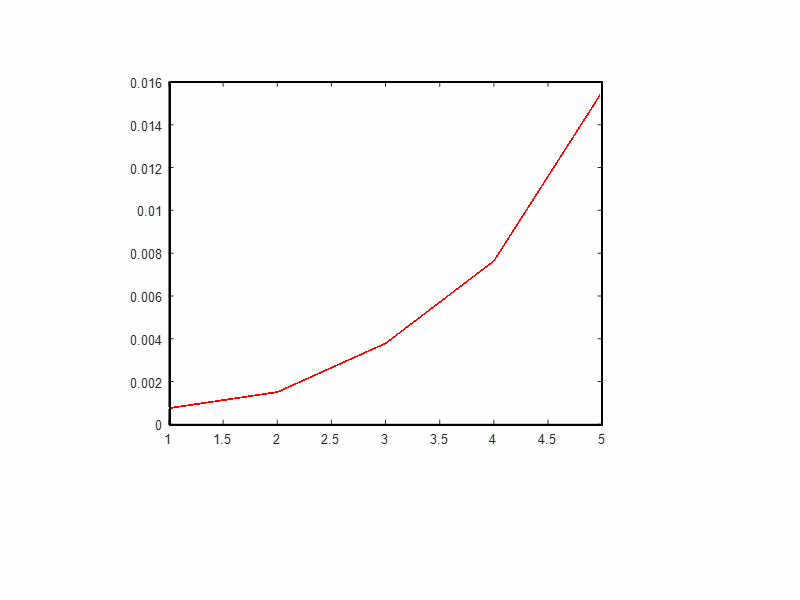
end

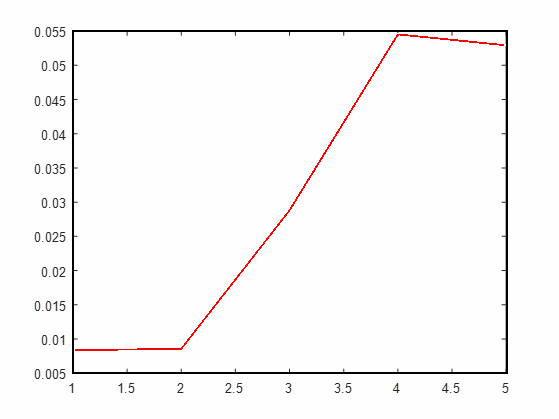
**Wykresy:**

Zaburzenie jednego elementu wektora – dobrze uwarunkowana

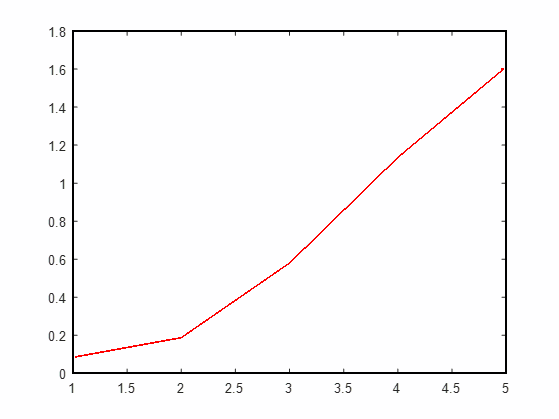


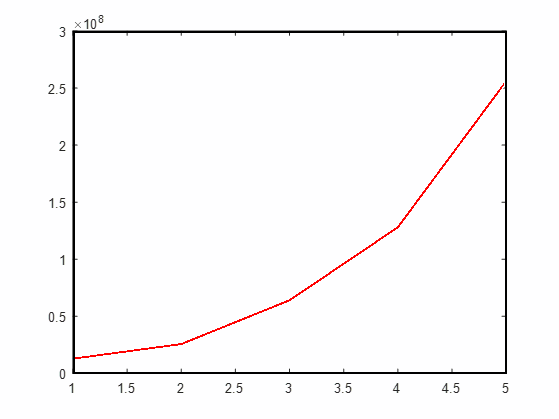
*Zaburzenie jednego elementu macierzy – dobrze uwarunkowana*

*Zaburzenie całej kolumny macierzy – dobrze uwarunkowana*

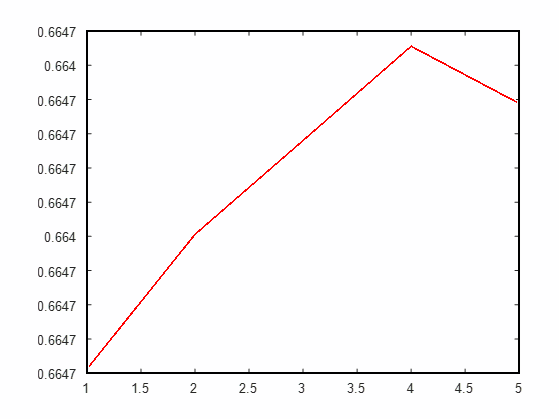


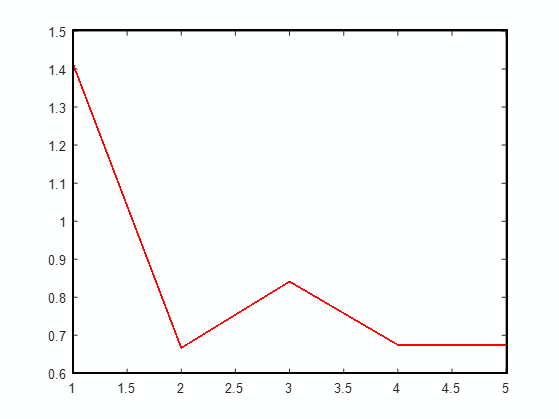
*Zaburzenie całego wektora – dobrze uwarunkowana*



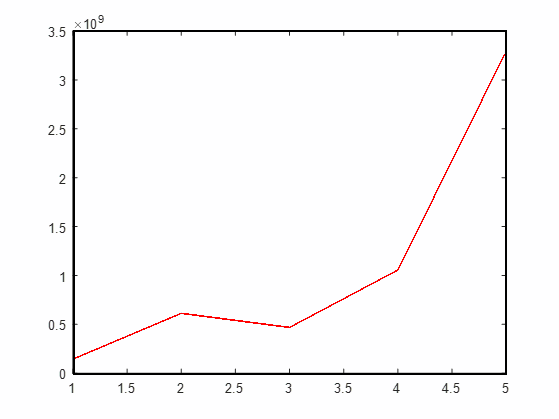
*Zaburzenie jednego elementu wektora – źle uwarunkowana*

*Zaburzenie jednego elementu macierzy – źle uwarunkowana*



*Zaburzenie całej kolumny macierzy – źle uwarunkowana*

*Zaburzenie całego wektora – źle uwarunkowana*



**Wnioski**

Czasy wykonywania się poszczególnych operacją wskazują, że najbardziej czuła na zmiany rozmiarów macierzy jest operacja odwracania macierzy. Najmniej wrażliwą na rozmiar macierzy operacją jest mnożenie danej macierzy przez dany wektor. Operacja zwracania śladu macierzy daje bardzo zbliżone wartości czasów potrzebnych na jej wykonanie dla wszystkich badanych rozmiarów macierzy.

Wraz ze wzrostem zaburzenia danych wejściowych wzrasta błąd względny wyniku.

Zaburzenia mają dużo większy wpływ w przypadku problemu źle uwarunkowanego, niż w przypadku problemu dobrze uwarunkowanego.

Zaburzenia wektora b(zarówno dla pojedynczego elementu jak i dla całości) mają większy wpływ na błędy wyniku niż zaburzenia macierzy A, w przypadku złego uwarunkowania.