

1. Gegeven de matrices A , B , C en I en de vector b . Hoe zou je in Matlab de vector x met

$$x = B^{-1} (2A + I) (C^{-1} + A) b$$

op een computationeel slimme manier berekenen ?

2. Op de achterzijde van dit blad staat `cp_11_06.m`, waarin o.a. de optimale waarde voor de SOR-parameter wordt gezocht. Dit gebeurt op een (computationeel gezien) domme manier. Kun je een betere oplossing voorstellen ?
3. Bepaal de interpolant $p_4(x)$ van graad 4 die $f(x) = x^3 - 4x + 6$ interpoleert in de Legendre-punten in het interval $[-2, 4]$.
4. Construeer een benaderingsformule voor $f''(x)$ op basis van de punten $(x + ih, f(x + ih))$, $i = -2, -1, 0, 1, 2$. Wat is de orde van deze formule ?
5. Beschouw het numerieke schema

$$\frac{y_j^{n+1} - y_j^n}{\Delta t} = \frac{y_{j+1}^{n+1} - 2y_j^{n+1} + y_{j-1}^{n+1}}{\Delta x^2}$$

voor het oplossen van de warmtevergelijking $u_t = u_{xx}$.

Gebruik de methode van Fourier om de stabiliteit van dit schema te onderzoeken.

```

function cp_11_06(n) % vergelijking van Poisson's vergelijking op L-vormig gebied

disp('directe methode');
% n=19;
h=1/(n+1);
n2=n^2;
b=h^2*2*ones(n2,1);
A=((diag(4*ones(n2,1))-diag(ones(n2-1,1),1)-diag(ones(n2-1,1),-1)-...
    diag(ones(n2-n,1),n)-diag(ones(n2-n,1),-n)));
for k=1:n-1 % 0 vanwege L-vormig gebied
    m=k*n;    A(m,m+1)=0;    A(m+1,m)=0;
end;
d=(n-1)/2;
el=(n-1)*n+1;
for k=d*n+1:n:el
    for m=0:d
        j=k+m; A(j,j-n)=0; A(j,j-1)=0; A(j,j+1)=0;
        if k~=el
            A(j,j+n)=0;
        end
        b(j)=0;
    end
end
u=A\b;
U=zeros(n+2,n+2);
U(2:n+1,2:n+1)=reshape(u,n,n);
v=0:h:1;
figure;
mesh(v,v,U); xlabel('x'); ylabel('y'); zlabel('U'); view(80,30);
title(sprintf('Computer probleem 11.6 - Directe methode, h=%4.3f',h));

disp('SOR methode');
D=diag(diag(A));
L=tril(A,-1);
U=triu(A,1);
options=optimset('TolX',1e-4,'TolFun',1e-4);
[omega,minim]=fminbnd(@spectral_radius,0,2,options,D,L,U)
x=zeros(n2,1);
resid=1;
tol=1e-4;
tel=0;
bb=omega*((D+omega*L)\b);
G=(D+omega*L)\((1-omega)*D-omega*U);
while resid > tol
    tel=tel+1;
    x= G*x+bb;
    resid=norm(b-A*x,inf);
end
X=zeros(n+2,n+2);
X(2:n+1,2:n+1) = reshape(x,n,n);
v=0:h:1;
figure;
mesh(v,v,X); xlabel('x'); ylabel('y'); zlabel('u'); view(80,30);
title(sprintf('Comp. probl. 11.6 - SOR methode, h=%4.3f omega=%4.3f iters=%4d rho=%4.3f',h,omega,tel,minim));

function [rho]=spectral_radius(omega,D,L,U)
G=(D+omega*L)\((1-omega)*D-omega*U);
rho=max(abs(eig(full(G))));

```