Kapittel 3

Bruk av Matlab

Oppgave 1

a) `>> sum(diag(A)) + sum(diag(B)) + sum(diag(C))`

eller:

```
>> sum(diag(A) + diag(B) + diag(C))
```

eller:

```
>> sum(diag(A + B + C))
```

b) `>> sum(A(1,:)) + sum(A(2,:))`

eller:

```
>> sum(sum(A(1:2,:)'))
```

c) `>> v = linspace(-4*pi,4*pi,17)`

d)
```
>> D = 3*eye(8) + diag(2*ones(7, 1), -1) + diag(2*ones(7, 1), 1) +
        diag(ones(6, 1), -2) + diag(ones(6, 1), 2)
```

Oppgave 2

a)
```
function n=lengde(x, y, z)
% Returnerer lengden til en vektor
% med komponenter {x,y,z}

if nargin < 3
  error('lengde tar 3 innparametre!')
end

n = sqrt(x^2 + y^2 + z^2);
end
```

b)
```
function n=lengde2(v)
% Returnerer lengden til en vektor v
% med vilkårlig dimensjon

n = sqrt(sum(v.^2));
end
```

Oppgave 3

a)
```
x = linspace(-2*pi,2*pi,401);
y = 4*sin(2*x);
plot(x, y)
```

b)
```
x = linspace(-10,2,12001);
y = x.^2.*exp(.5*x);
plot(x, y)
```

c)
```
t = linspace(0,8*pi,401);
x = t.*cos(t);
y = t.*sin(t);
plot(x, y)
```

d)
```
t = linspace(-pi,pi,63);
[x,y] = meshgrid(t);
z = sin(x).*sin(y);
surf(x, y, z)
```

e)
```
t = linspace(1,5,401);
[x,y] = meshgrid(t);
z = log(x.*y)./(x.^2 + y.^2);
[C,h] = contour(x, y, z);
clabel(C, h)
```

f)
```
[x,y] = meshgrid(-5:1:5);
u = x./(x.^2 + y.^2);
v = y./(x.^2 + y.^2);
figure(1)
quiver(x, y, u, v)
figure(2)
quiver(x, y, u, -v)
figure(3)
quiver(x, y, v, u)
figure(4)
quiver(x, y, v, -u)
```

Oppgave 4

a)
```
[x,y] = meshgrid(linspace(0,2,21), linspace(-2,2,41));
h = 1000 + 50*x.^2.*y.^2.*exp(1 - x.^2);
surf(x, y, h, 'FaceColor', [.36 .67 .93])
```

```
axis square
xlabel('x')
ylabel('y')
zlabel('h')
```

b)
```
[x,y] = meshgrid(linspace(0,2,21), linspace(-2,2,41));
h = 1000 + 50*x.^2.*y.^2.*exp(1 - x.^2);

% La Matlab velge høydenivåer:
%[C,h] = contour(x, y, h);

% Plot et gitt antall høydenivåer:
%[C,h] = contour(x, y, h, 20);

% Plot bare noen utvalgte høydenivåer:
v = 1001:10:1091;
[C,h] = contour(x, y, h, v);

clabel(C, h)
colorbar
axis square
```

c)
```
[x,y] = meshgrid(linspace(0,2,21), linspace(-2,2,41));
h = 1000 + 50*x.^2.*y.^2.*exp(1 - x.^2);

v = 1001:10:1091;
contour(x, y, h, v);
colorbar
axis square

[x,y] = meshgrid(linspace(0,2,11), linspace(-2,2,21));
h = 1000 + 50*x.^2.*y.^2.*exp(1 - x.^2);
[dhx,dhy] = gradient(h, 0.2);  % avstand mellom gridpunktene er 0.2
hold on
quiver(x, y, dhx, dhy, 0.5)
hold off
```

Oppgave 5

a)
```
function f=fakultet(n)
% f=fakultet(n) returnerer n!

f = prod(1:n);
end
```

b)
```
function f=funk(x, n)
% f=funk(x, n) returnerer den n'te deriverte til cos(x)

if mod(n, 4) == 0,
  f = cos(x);
```

```
elseif mod(n, 4) == 1,
  f = - sin(x);
elseif mode(n, 4) == 2,
  f = -cos(x);
elseif mod(n, 4) == 3,
  f = sin(x);
end
end
```

eller:

```
function f=funk(x, n)
% f=funk(x, n) returnerer den n'te deriverte til cos(x)

f = cos(x + n*pi/2);
end
```

c)
```
function f=taylor1D(x, x0, n)

f = funk(x0, 0);
for i=1:n,
  f = f + (1/fakultet(i))*funk(x0, i)*(x - x0)^i;
end
end
```

d)
```
N = 51;
x = linspace(-pi/2,3*pi/2,N);
y = cos(x);
plot(x, y, 'r');
axis equal
hold on

x0 = pi/2;
y1 = []; y2 = []; y3 = []; y4 = [];

for i=1:N,
  y1(i) = taylor1D(x(i), x0, 1);
  y2(i) = taylor1D(x(i), x0, 3);
  y3(i) = taylor1D(x(i), x0, 5);
  y4(i) = taylor1D(x(i), x0, 7);
end

plot(x, y1, 'b');
plot(x, y2, 'g');
plot(x, y3, 'c');
plot(x, y4, 'm');
hold off
```