

Johdatus Matlabin käyttöön Syksy 2015, I periodi

Tehtäviä 2, 4.9.2015

henrik.kettunen@helsinki.fi

```
% Tehtäviä 2, 16.1.2015 Ratkaisuja

% TEHTÄVÄ 1

x1 = [1 2 3 4 5]
% tai x1 = 1:5
% tai x1 = linspace(1,5,5)

x2 = 2:2:100

% 'Askel' alkioiden välillä määrättävä erikseen negatiiviseksi
x3 = 5:-1:-5

x4 = 1:10
% muutetaan viimeisen (10.) alkion arvo
x4(10) = 11

x5 = 5*ones(100,1)

x6 = zeros(1,20)
% korvataan parilliset alkiot arvolla 1
x6(2:2:20) = 1

% Yhdistetään kaksi vektoria peräkkäin
x7 = [1:2:99, 2:2:100]

x8 = 1:100
% sijoitetaan parillisiin alkioihin uudet arvot
x8(2:2:100) = 100:-2:2

x9 = 0:10
% Korotetaan vektori alkioittain toiseen potenssiin
x9 = x9.^2

x10 = linspace(0,100,20)
% tai x10 = 0:100/19:100

% Kuten x6
x11 = ones(1,100)
x11(2:2:100) = 2

% Kyseessä vektori 1:100, jonka parilliset alkiot on korotettu toiseen
% potenssiin.
x12 = 1:100
% Korotetaan vektori alkioittain potensseihin, jotka määräytyvät vektorin
% x11 mukaan
x12 = x12.^x11
% Tai sitten korotetaan vain vektorin parilliset alkiot toiseen potenssiin
x12 = 1:100
x12(2:2:100) = x12(2:2:100).^2

x13 = exp(1:10)

% TEHTÄVÄ 2

A1 = [1 2; 3 4]

A2 = zeros(10)
% tai A2 = zeros(10,10)
```

```

A3 = 7*ones(5,2)

A4 = ones(4,5)
% korvataan valitut alkiot
A4(1,2) = 8
A4(4,3) = -4
A4(2,5) = 0

A5 = zeros(5)
% korvataan matriisiin 3. sarake ja 3. rivi
A5(:,3) = 1:5
A5(3,:) = 5:-1:1

A6 = eye(8)

A7 = diag(1:5)
% tai A7 = diag([1 2 3 4 5])

% Muodostetaan ensin vektori, joka järjestetään uudelleen matriisiksi
A8 = 1:25
A8 = reshape(A8,5,5)
% Tarvitaan vielä transpoosi
A8 = A8.'

A9 = 1:10
A9 = repmat(A9,10,1)
% tai kahden vektorin avulla
A9 = ones(10,1)*(1:10)
% tai kahden matriisin avulla
A9 = ones(10)*diag(1:10)

% Käännetään matriisiin A9 parilliset rivit ympäri
A10 = A9
A10(2:2:10,:) = fliplr(A10(2:2:10,:))
% jonka jälkeen vielä transponoidaan matriisi
A10 = A10.'

% TEHTÄVÄ 3

a = [1 0 3]
b = [2 4 5]
c = [9 3 1]

dot(a,b)
dot(b,a)
dot(c,a)
cross(a,b)
cross(b,a)
cross(a,c)
dot(cross(c,a),b)

% TEHTÄVÄ 4

% Pistetulo
a*b.'
% tai
sum(a.*b)
% tai
a(1)*b(1) + a(2)*b(2) + a(3)*b(3)

% Ristitulo on laskettava ihan komponenteittain
[a(2)*b(3)-b(2)*a(3), a(1)*b(3) - b(1)*a(3), a(1)*b(2)-b(2)*a(1)]

% LISÄTEHTÄVÄT 5

```

```

M = [3 1; 1 3]

det(M)
inv(M)

[V,D] = eig(M)
% Matriisin V sarakkeet ovat matriisin M ominaisvektorit ja matriisin D
% diagonaalialkiot ovat näitä vastaavat ominaisarvot

% poimitaan ominaisarvot
lambda1 = D(1,1)
lambda2 = D(2,2)

% poimitaan ominaisvektorit
v1 = V(:,1)
v2 = V(:,2)

% Tarkistetaan laskemalla, toteutuuko M*v - lambda*v = 0
M*v1 - lambda1*v1
M*v2 - lambda2*v2

%%%%%%%%%%

% Kootaan matriisi paloittain
N = [[1 2; 3 2], zeros(2); zeros(2), [3 1; 1 3]]

det(N)
inv(N)

[V,D] = eig(N)

% poimitaan ominaisarvot (4 kpl!)
lambda1 = D(1,1)
lambda2 = D(2,2)
lambda3 = D(3,3)
lambda4 = D(4,4)

% poimitaan ominaisvektorit
v1 = V(:,1)
v2 = V(:,2)
v3 = V(:,3)
v4 = V(:,4)

% Tarkistetaan laskemalla, toteutuuko M*v - lambda*v = 0
N*v1 - lambda1*v1
N*v2 - lambda2*v2
N*v3 - lambda3*v3
N*v4 - lambda4*v4

```