

- Perehdytään omien ohjelmien (script, m-file) kirjoittamiseen Matlabin editorin avulla. Luo omaan kotihakemistoon tiedostonhallinnan kautta tätä kurssia varten uusi alihakemisto esim. 'Matlab-kurssi', ja siirry Matlabissa tähän hakemistoon. Avaa Matlabin editori valitsemalla yläreunasta New -> Script

- Hello World!

- » Kirjoita editori-ikkunaan rivi
`disp('Hello World!')`
ja tallenna tiedosto esim. nimellä hello.m
- » Siirry takaisin Command Window'n puolelle ja kirjoita komentoriville
`hello`
(eli äsken tallentamasi tiedoston nimi ilman .m-päätettä)
- » Nauti lopputuloksesta.

- Luo editorilla uusi tiedosto (Create New Document). Kirjoita (tai kopioi) siihen seuraavat rivit
%-alkuiset rivit ovat kommentteja, joita Matlab ei noteeraa
(Copy-Pasten kanssa '-merkki saattaa muuttua vääräksi, joten se on korjattava erikseen)

```
% Tyhjennetään kaikki muuttujien arvot  
clear all;
```

```
N = 50; % Laskentapisteiden määrä  
Ac = 1; % Kosinin amplitudi  
As = 2; % Sinin amplitudi  
mc = 2; % Kosinin 'taajuus'  
ms = 1; % Sinin 'taajuus'
```

```
% Luodaan vektori, jonka pisteissä  
% funktioiden arvot lasketaan ja piirretään  
x = linspace(-pi,pi,N);
```

```
% Lasketaan funktiot  
y1 = Ac*cos(mc*x);  
y2 = As*sin(ms*x);
```

```
% Piirretään kuva  
figure(1); plot(x,y1,'b*',x,y2,'r:','LineWidth',1.5); grid;  
xlabel('x') % x-akselin otsikko  
ylabel('f(x)') % y-akselin otsikko  
legend('kosini','sini') % Kuvaajien selitteet
```

- » Tallenna tiedosto esim. nimellä 'piirra_cos_ja_sin.m'
- » Aja ohjelma komentoriviltä: `piirra_cos_ja_sin`
- » Käy tämän jälkeen editorissa muuttelemassa eri parametreja, muista aina tallentaa tiedosto uudeen, ja aja se jälleen komentoriviltä
- » Havainnoi, miten kuvaaja muuttuu
- » Kokeile myös muuttaa kuvaajan värejä, esim 'b', 'r', 'g', 'c', 'y', 'k' ja piirtotyylejä, kuten '--', '.', ':', '*', '+', '0-

- Seuraava koodinpätkä visualisoi kaksiulotteista paloittain vakiota pyörähdyssymmetristä korkeusjakaumaa

$$f(r) = \begin{cases} 1, & r < 0.25 \\ 0.5, & 0.25 \leq r < 0.75 \\ 0 & r \geq 1 \end{cases}$$

```
% Siivotaan kaikki muuttujat
clear all

% Määrätään laskenta- ja piirtoalueen rajat
x = linspace(-1.5,1.5,101);
% Samat rajat x- ja y-suunnassa
y = x;

% Luodaan kaksiulotteinen laskentahila
[X,Y] = meshgrid(x,y);

% Säteittäinen etäisyys origosta
R = sqrt(X.^2 + Y.^2);

% Alustetaan matriisi Korkeus oikean kokoiseksi ja täyteen nolliä
% Huom! Tämän jälkeen korkeuden nollatasoa ei tarvitse erikseen laskea.
Korkeus = zeros(size(R));

% Etsitään find-komennolla matriisista 'R' ne alkiot, jotka
% toteuttavat oikeat ehdot, eli niiden etäisyys origosta on
% tietyllä välillä.
alue1 = find(R < 0.25);
alue2 = find(R >= 0.25 & R < 0.75);

% Sijoitetaan Korkeus-matriisiin näitä vastaaviin alkioihin
% oikea korkeusdata
Korkeus(alue1) = 1;
Korkeus(alue2) = 0.5;

% Piirretään hieno 3D-kuvaaja
figure(1); surf(X,Y,Korkeus)
```

» Kirjoita tai kopioi oheiset rivit editoriisi ja tallenna tiedosto esim. nimellä 'Korkeus.m'. Tutki, mitä milläkin rivillä tehdään. Perehdy erityisesti find-komennon toimintaan.

- Osaatko tämän jälkeen lisätä torniin pari kerrosta, eli muokata tiedostoa siten, että se esittää korkeusjakaumaa

$$f(r) = \begin{cases} 1, & r < 0.25 \\ 0.75, & 0.25 \leq r < 0.5 \\ 0.5, & 0.5 \leq r < 0.75 \\ 0.25, & 0.75 \leq r < 1 \\ 0, & r \geq 1 \end{cases}$$

» Huom! Ennen muokkausta tallenna tiedosto uudelleen eri nimellä (Save As) esim. 'Korkeus2.m'