

1. if-rakenne. Kirjoita seuraava kolikonheittoa simuloiva skripti. (Tallenna esim. nimellä 'Kolikko.m')

```
heitto = randi(2);
if heitto == 1
    disp('Kruuna')
else
    disp('Klaava')
end
```

2. Perehdy for-silmukan toimintaan seuraavien esimerkkien avulla.

- Esimerkki 1

```
for a = 1:10
    b = a^2;
    pause; % Keskeyttää ajon, kunnes käyttäjä painaa mitä tahansa näppäintä
    disp([a, b])
end
```

- Esimerkki 2

```
x = [5 6 -7.5 0 2 2 3.212];
N = length(x);
for n = 1:N
    % Jotta disp-käskey osaisi lukea auki muuttujien arvoja, on ne ensin
    % muutettava merkkijonoiksi
    disp([int2str(n), '. luku on ', num2str(x(n))])
end
```

3. Perehdytään seuraavaksi omien funktioiden kirjoittamiseen. Avaa editori New -> Script (Tai suoraan New -> Function)

- Seuraava funktio piirtää yksiulotteisen Gaussin käyrän käyttäjän antamien parametrien (laskentapisteet  $x$  (vektori), maksimiarvo  $A$  (skalaari), odotusarvo  $x_0$  (skalaari) ja keskihajonta  $\sigma$  (skalaari)) avulla. Ts.  $g(x) = Ae^{-\frac{(x-x_0)^2}{2\sigma}}$ .

- Kirjoita seuraavat rivit editori-ikkunaan ja tallenna tiedosto omaan kurssihakemistoosi nimellä 'piirra\_Gauss.m'

```
% Funktion määrittely:
% g = piirra_Gauss(x,A,x0,sigma)
% Funktio laskee ja piirtää Gaussin käyrän seuraavilla syöttöparametreilla:
% x: laskentapisteet (vektori)
% A: funktion maksimi (skalaari)
% x0: odotusarvo (skalaari)
% sigma: keskihajonta (skalaari)
% Funktio palauttaa:
% g: lasketut käyrän arvot pisteissä x
```

---

```

function g = piirra_Gauss(x,A,x0,sigma)

% Funktion arvojen laskeminen
g = A*exp(-(x-x0).^2/(2*sigma));

% Funktion kuvaajan piirto
figure(1); plot(x,g); grid
end

```

- Mitä tapahtuu, kun nyt pyydät Matlabilta 'help piirra\_Gauss'?

4. Oletetaan, ettei Matlabissa olisi valmiita komentoja 'dot' ja 'cross' vektorien piste- ja ristitulon laskemiseksi, joten kirjoita näiden sijaan omat funktiot

```

function c = pistetulo(a,b)
ja
function c = ristitulo(a,b)

```

jotka suorittavat nämä operaatiot. Funktion syöttöparametrit a ja b ovat 3-pituisia vektoreita. Funktio 'pistetulo' palauttaa skaalarin c, ja 'ristitulo' vektorin c. Testaa vielä, että omat funktiosi antavat saman tulokset kuin Matlabin omat 'dot' ja 'cross'!

5. Kirjoita funktio `[kruunat,klaavat] = kolikko_pie(N)`, joka heittää noppaa käyttäjän määräämät N kertaa ja palauttaa tiedon esiintyneiden kruunien (funktion arvo 'kruunat') ja klaavojen (funktion arvo 'klaavat') määräästä ja raportoi joka heitosta yksi kerrallaan, kumpi tuli tulostamalla ruudulle joko tekstin 'Kruuna' tai 'Klaava'. Aseta heittojen välille 0.5 sekunnin viive ('help pause') jännitystä lisäämään! Toteuta funktio for-silmukan avulla. Lopuksi funktio piirtää piirakkadiagrammin ('help pie') kruunien ja klaavojen jakaumasta.
6. Kirjoita funktio `even_odd(x)`, joka käy läpi käyttäjän antaman vektorin x sisältämät luvut yksi kerrallaan ja tutkii ja kertoo, ovatko syötetyt luvut kokonaislukuja, ja jos ovat, ovatko ne parillisia vai parittomia. Funktio raportoi jokaisesta vektorin alkioista  $x_n$  joko ' $x_n$  on pariton kokonaisluku', ' $x_n$  on parillinen kokonaisluku' tai ' $x_n$  ei ole kokonaisluku'. Muuttujan  $x_n$  tilalle siis tietysti tulostuu kyseisen luvun arvo ('help disp', 'help num2str'). Onko luku kokonaisluku, kuten myös sen parillisuus/parittomuus, voidaan selvittää funktion 'mod' (modulus eli jakojäännös) avulla ('help mod').