Ett exempel med tärningskast och ledning till lab

Vi vill simulera kast med en tärning och räkna frekvenserna av ettor, tvåor etc. Här följer en sekvens av funktioner. Antalet kast bestäms av en inparameter **num_throws**.
Här är den första lösningen, som är rätt opraktisk.

```
function die_freq1(num_throws)

a = 0; b = 0; c = 0; d = 0; e = 0; f = 0;
for k = 1:num_throws
  throw = rand;
  if throw < 1/6
    a = a + 1;
  elseif throw < 2/6
    b = b + 1;
  elseif throw < 3/6
    c = c + 1;
  elseif throw < 4/6
    d = d + 1;
  elseif throw < 5/6
    e = e + 1;
  else
    f = f + 1;
  end
end

a, b, c, d, e, f
```

```
>> die_freq1(10000)
a = 1656
b = 1674
c = 1660
d = 1638
e = 1701
f = 1671
```


Lösningen är opraktiskt att programmera och opraktiskt för användare (människor och program) av funktionen. Man kan inte enkelt efterbehandla, t.ex. plotta, frekvenserna. Tänk om sinus-funktionen skrev ut resultatet, då skulle *inte* följande fungera

```
>> r = 1.23;
>> x = r * cos(0.1);
>> y = r * sin(0.1);
```

Följande lösning, **die_freq2** är bättre, men ändå lite opraktisk. Jag har kopierat **die_freq1** men tagit bort utskriftsraden.
Första raden i **die_freq2** lyder:

```
function [a, b, c, d, e, f] = die_freq2(num_throws)
```

och så här används funktionen:

```
>> [a, b, c, d, e, f] = die_freq2(10000)
a = 1614
b = 1706
c = 1667
d = 1623
e = 1682
f = 1708
```

Anropet är lite jobbigt: Notera att följande inte gör det man kanske tror:

```
>> frekvens_vektor = die_freq2(10000)
frekvens_vektor = 1636
```

Man får bara första frekvensen **a**. Analogt ger

```
>> [a, b] = die_freq2(10000)
```

de två första frekvenserna.

Det är enklare för alla parter om man förpackar frekvenserna i en vektor, **freqs**, så här:

```
function freqs = die_freq3(num_throws)

freqs  = zeros(6, 1);

for k = 1:num_throws
  throw = rand;
  if throw < 1/6
    freqs(1) = freqs(1) + 1;
  elseif throw < 2/6
    freqs(2) = freqs(2) + 1;
  elseif throw < 3/6
    freqs(3) = freqs(3) + 1;
  elseif throw < 4/6
    freqs(4) = freqs(4) + 1;
  elseif throw < 5/6
    freqs(5) = freqs(5) + 1;
  else
    freqs(6) = freqs(6) + 1;
  end
end
```

man kan nu enkelt hantera resultatet:

```
>> frekvens_vektor = die_freq3(10000)
frekvens_vektor =
        1675
        1655
        1628
        1758
        1645
        1639
>> (frekvens_vektor - 10000 / 6)'
ans = 8.3333 -11.6667 -38.6667  91.3333 -21.6667 -27.6667
```


Med vektorer kan man också skriva enklare kod:

```
function freqs = die_freq4(num_throws)
freqs  = zeros(6, 1);

for k = 1:num_throws
  throw = rand;
  for j = 0:5
    if j/6 <= throw & throw < (j+1)/6
      freqs(j+1) = freqs(j+1) + 1;
      break
    end
  end
end
```

Detta går att göra *ännu kortare* (man kan bli av med if-satsen och for-j-loopen), men det visar jag inte (det skall vara något kvar till labben också).

När man skall redovisa en sådan uppgift i en laboration, skriver man lämpligen ett huvudprogram, **die_main** säg, som kan vara en script-fil (behöver inte vara en funktion). Kanske något i stil med.

```
n_throws = 1000; % antalet tärningsskast

% Här testar vi ...
dice_freq1(n_throws)
% och här
[a, b, c, d, e, f] = dice_freq2(n_throws)
freqs_3 = dice_freq3(n_throws)
freqs_4 = dice_freq3(n_throws)

% Lite analys av...
freqs_4 - n_throws / 6

% Nu plottar vi ...
plot(...
```

En annan lab-fråga är hur man presenterar resultatet. Här är en primitiv variant. Mer om detta står i lab-pm.

```
>> frekvens_vektor = die_freq3(10000);
% skapa en matris med två kolonner
>> [(1:6)', frekvens_vektor]
ans =
           1        1679
           2        1665
           3        1627
           4        1661
           5        1684
           6        1684

% skriver inte ut ans
>> disp([(1:6)', frekvens_vektor])
           1        1679
           2        1665
           3        1627
           4        1661
           5        1684
           6        1684
```

Så, i huvudprogrammet kan man ha något i stil med:

```
disp('Resultat av tärningsskast.')
disp('         Ögon     Frekvens')
disp([(1:6)', frekvens_vektor])
```

som ger

```
Resultat av tärningsskast.
         Ögon     Frekvens
           1        1682
           2        1733
           3        1599
           4        1640
           5        1704
           6        1642
```