
Matematikk 1000

Øvingsoppgaver i numerikk – leksjon 3

Løsningsforslag

Oppgave 1 – Flo og fjære

a) `>x=0:.1:24;`

`>y=3.2*sin(pi/6*(x-3));`

Disse linjene burde vel være forståelige nå.

`>plot(x,y,'linewidth',3)`

Grafen får tykkelsen 3 (“default” er 1).

`>set(gca,'fontsize',20)`

Justerer opp skriftstørrelsen.

`>xlabel('t [timar]'); ylabel('T [meter]')`

Setter tekst på aksene (både x - og y -aksen).

`>hold on`

`>plot([0 24],[-1 -1],'k-')`

Plotter linja $y = -1$ for x fra og med 0 til og med 24.

`>plot([6 18],[3.2 3.2],'ro','linewidth',2)`

`>plot([2.39 9.61 14.39 21.61],[-ones(1,4),'gd'])`

Her plotter vi inn diverse punkt - de første to blir markert med en (tykk) rød (r) sirkel (o), og de fire neste blir markert med grønne (g) “diamanter” (d). Det viser seg forresten at ‘d’ ikke gir noen “diamant” for alle versjoner av MATLAB. For å finne ut mer om hvilke muligheter man har (farger, typer punkt etc.), kan man skrive ‘>help plot’.

`>legend('T(t)', 'y=-1', 'Flo', 'T(t)=-1')`

legend-kommandoen gjør oss i stand til å forklare hva de ulike grafene og punktene representerer. “Input”-variablene til legend-, xlabel og ylabel-funksjonene er små tekst-snutter inni apostroffer¹. Slike små tekst-

¹Hermetegn kan også brukes

variabler kalles *strenger*.

```
>axis([0 27 -4 4])
```

Her justerer vi x -aksen litt ut slik at grafen ikke kolliderer med teksten i hjørnet øverst til høyre.

b) Om vi tar med kommentarer, kan skriptet se slik ut:

```
% Skript som plotter en modell for tidevann
x=0:.1:24;                                % Vektor med argument-verdier
y=3.2*sin(pi/6*(x-3));                    % Vektor med fusjonsverdier
plot(x,y,'k','linewidth',3)              % Plotter funksjonen
set(gca,'fontsize',20)                   % Fikserer skriftsstørrelsen på aksene
xlabel('t [timer]'); ylabel('T [meter]') % Setter tekst på aksene
hold on
plot([0 24],[-1 -1],'k-')                % Markerer linja y=-1
plot([6 18],[3.2 3.2],'ro','linewidth',2) % Markerer punktene for flo
plot([2.39 9.61 14.39 21.61],[-ones(1,4),'gd']) % Markerer når vannstanden var -1 m
legend('T(t)', 'y=-1', 'Flo', 'T(t)=-1') % Forklarende tekst til grafene
axis([0 27 -4 4])                        % Justerer intervallene aksene går over
hold off
```

Som sagt er det en god vane å kommentere skript, selv om det kanskje er gjort litt vel omstendelig her.

Oppgave 2 – Summer og for-løkker

a)

$$\sum_{i=1}^{10} i^2 = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + 6^2 + 7^2 + 8^2 + 9^2 + 10^2 = 1 + 4 + 9 + 16 + 25 + 36 + 49 + 64 + 81 + 100 = 385 \quad .$$

c) I kommandovinduet får vi opp følgende:

```
>FinnSum
sum = 385
```

Dette stemte jo bra. Om vi dropper semikolonet i linje 3, får vi

```
>FinnSum
sum = 1

sum = 5
```

```

sum = 14

sum = 30

sum = 55

sum = 91

sum = 140

sum = 204

sum = 285

sum = 385

sum = 385

```

- c) Skriptet kan lett tilpasses ved å endre 10 til 100 i linje 2:

```

sum=0;
for i=1:100
    sum=sum+i^2;
end
sum

```

Det er lurt allerede nå å venne seg til å lagre fila med hurtigtast, typisk **Ctrl+S**, i stedet for å klikke i menyen; ellers kan det kan fort bli i overkant mye klikking. Om vi nå kjører skriptet, får vi svaret 338350.

- d) Nedre grense skal nå være 5, og øvre grense skal være 20. Hvor hver gang legger vi til $\sqrt{i}$, som vi skriver som `'sqrt(i)'` i MATLAB:

```

sum=0;
for i=5:20
    sum=sum+sqrt(i);
end
sum

```

Vi kjører skriptet og får (det tilnærma) svaret 55.5197.

- e) I linje 2 ser vi at `i` går, i steg på 1, fra og med 0 til og med 12. Dette er altså summasjonsgrensene. I linje 3 ser vi at for hver omgang, hver *iterasjon*, blir det lagt til $1/i$. Summen er altså

$$\sum_{i=0}^{12} \frac{1}{i} \quad .$$

Vi kommer tilbake til summer av denne typen når vi senere i kurset skal beregne integraler.