

---

# Matematikk 1000

## Øvingeoppgaver i numerikk – leksjon 1

### *Å komme i gang*

---

I denne øvinga skal vi bli litt kjent med MATLAB. Vi skal ikkje gjøre noen avanserte ting i dette oppgavesettet – bare få et visst innblikk i overflata av hva MATLAB kan gjøre.

Det at det er mye tekst i oppgavesettet betyr ikke nødvendigvis at det er mye å gjøre. En del av dere vil nok oppleve at instruksjonene i disse oppgavene er mer detaljerte en strengt tatt nødvendig. Grunnen til dette er at vi ønsker at det tekniske skal by på minst mulig utfordringer – også for de som ikke har brukt datamaskiner på denne måten tideligere. Så vi håper du bærer over med “detaljstyringa”, eventuelt.

Meningen med disse oppgavene er å pirre nysgjerrigheten din på om hva vi kan bruke MATLAB til. Dersom nysgjerrigheten din drar deg i litt andre retninger enn det disse oppgavene forsøker å gjøre, så er det helt greit.

Til slutt nevner vi at det finnes flere MATLAB-kloninger som er gratis. Vi kan for eksempel nevne Octave, SciLab og Freemath. Disse kan det være greit å vite om den dagen du ikke lenger har MATLAB-lisens gjennom HiOA.

### Oppgave 1 – Få MATLAB opp å gå

MATLAB kan lastes ned fra nettsida til MathWorks. På nettsida til kurset finner du notat som forklarer hvordan dette kan gjøres.

Installér MATLAB på datamaskina di ved å følge instruksjonene nøye. Dette bør ta omtrent en halv time. Ikke nøl med å be om hjelp dersom det skulle oppstå problemer.

Når installasjonen er vel gjennomført, start opp MATLAB. Ved oppstart vil et panel med fire viduer dukke opp:

- Mappevinduet, “*Current Folder*”, som viser hvilken mappe du jobber i.

- Kommandovinduet, “*Command Window*”, hvor en utfører de ulike kommandoane. Dette er det viktigste vinduet, og det går fint an å bruke MATLAB med dette vinduet alene.
- Variabelvinduet, “*Workspace*”, som viser hvilke variable som ligger i minnet.
- Kommandohistorien, “*Command History*” – ei liste som viser alle kommandoene du har utført i arbeidsøkta.

I dette oppgavesettet skal vi stort sett bare prate om kommandovinduet.

## Oppgave 2 – MATLAB som kalkulator

Her skal vi kort forklare hvordan man utfører enkle aritmetiske operasjoner i MATLAB. Dette er definitivt å “skyte spurv med kanon”, men vi må jo begynne et sted.

- a) Forsøk å skrive inn enkle aritmetiske regnestykker i kommandovinduet. Du kan for eksempel prøve å regne ut

```
2+2
3-2
3*2
6^2
6/2
```

Hva betyr ‘\*’, ‘^’ og ‘/’ i denne sammenhengen?

- b) Regn ut

$$\frac{3^{2.12} \cdot (2 + 4 \cdot (-3.1))}{2^2 - 1}.$$

Gjør gjerne flere – og mer avanserte – regnestykker.

Som du sikkert ser, er ‘\*’ symbolet som blir brukt for multiplikasjon, ‘/’ blir brukt for divisjon og ‘^’ blir brukt for å skrive potenser<sup>1</sup>.

- c) Hva blir regna ut her?:

```
sin(0.7)
log(5)
sqrt(9)
exp(1)
atan(2)
```

---

<sup>1</sup>Hatten, ‘^’, kan være litt “krøkkete” å få fram på visse tastatur. Det hender man må kombinere den med mellomrom-tasten.

d) Finn ut hvordan MATLAB tolker disse regnestykkene:

```
1/0
0^0
5/Inf
0*Inf
Inf/Inf
10^999
1.2e4
NaN+1
```

Er du enig i alle disse tolkningene; virker de rimelige? ‘Inf’ betyr uendelig,  $\infty$ , (*infinty*), og ‘NaN’ betyr ‘undefinert’ (*Not a number*). Som du sikkert ser, betyr ‘1.2e4’  $1.2 \cdot 10^4$  (tall skrevet på *normalform*).

e) Noen tall er så spesielle at de har fått egne navn. I kommandovinduet, skriv

```
pi
i
eps
```

‘eps’ refererer her til den greske bokstaven epsilon,  $\epsilon$ , som ofte blir brukt for å symbolisere små tall. Dette tallet gir størrelsen på det minste tallet  $x$  som er slik at maskina klarer å skille mellom 1 og  $1 + x$ . Vi kaller dette ‘*maskinpresisjonen*’ og, som navnet tilsier, varierer denne presisjonen noe fra maskin til maskin.  $i$  blir kalt ‘den imaginære enheten’ og kan oppfattes som  $\sqrt{-1}$  (joda, vi *kan* tillate oss å ta kvadratrota av minus én; dette kommer vi snart tilbake til).

### Oppgave 3 – Tilordning

a) Vanligvis vil vi ha behov for å operere med flere tallstørrelser samtidig. Vi kan *tilordne* tallverdier til variabelnavn slik:

```
>> x=7
```

Symbolene “»” har vi her brukt for å indikere at dette blir skrevet inn i kommandovinduet i MATLAB. Vi har nå sagt at variabelen  $x$  skal ha verdi 7; dette “husker” MATLAB – det ligger i minnet. Variabelen  $x$  dukker opp i variabelvinduet (*workspace*). Vi kan så bruke denne variabelen i andre regnestykker. Regn ut

```
>> x*2
>> x^2
```

Vi kan fortsette å tilordne flere variable:

```
>> y=2
>> x/y
```

Etterhvert som du tilegner flere variable, vil også de dukke opp i lista i variabelvinduet. (Du kan også få opp en liste over alle variable i kommandovinduet ved å skrive `'who'`.)

Når vi gjør en utregning uten å gi svaret noe navn, slik vi gjorde i oppgave 1, vil svaret automatisk få navnet `ans` (for *answer*).

- b) Når vi bruker symbolet '=' som i forrige del-oppgave, betyr det slett ikke "er lik", som i matematisk notasjon. Her er likhetstegnet brukt som **tilordning**; vi putter tallet 7 inn i variabelen `x`. Det følgende gir faktisk mening i MATLAB:

```
>> x=2
>> x=x+1
```

Hva får vi til svar om vi gjør denne operasjonen? Hvorfor er den siste linja meningsløs om vi skal tolke den matematisk?

- c) Undersøk hva som skjer om du utfører følgende:

```
>> pi
>> pi=10
```

Hvilken verdi vil variabelen `pi` få?

**Litt om variabelnavn:** I utgangspunktet kan vi gi variable hvilket navn som helst; navnet kan godt bestå av flere bokstaver. Men man må huske visse ting:

- Et variabelnavn får ikke begynne med et tall
- Variabelnavn får ikke inneholde mellomrom eller parenteser.
- MATLAB skiller mellom store og små bokstaver
- Forsøk å unngå variabelnavn som er like kommandoer eller funksjonsnavn som finnes i MATLAB fra før – som foreksempel de funksjonene vi så i oppgave 2 c) (`'sin'`, `'atan'` o.s.v.).

Du avslutter MATLAB ved å klikke på 'quit' under nedtrekksmenyen oppe til venstre eller ved å skrive `exit` eller `quit` i kommandovinduet.