

Notat 1, del 5 (av totalt 9 deler)

Brukerstyrt innlesing (input) og utskrift (output)

Brukerdefinert innlesning (input)

- Fram til nå har vi definert variabelverdier i m-filen
- **input**-funksjonen tillater oss å legge inn en verdi mens MATLAB-programmet kjører

```
1 - z = input('Tast inn en verdi: ')
```

```
>> kap7
```

```
fx Tast inn en verdi:
```

Kommandovinduet blir aktivt, vi legger inn et tall her og trykker «Enter»-tasten.

```
1 - z = input('Tast inn en verdi: ')
```

Name	Value	Class
z	5	double

```
>> kap7
```

```
Tast inn en verdi: 5
```

```
z =
```

```
5
```

```
fx >>
```

input-funksjonen kan ta

- skalar (tall)
- matriser/ vektorer
 - Tallene må angis innenfor [og]
 - Mellomrom eller komma deler tall i samme rad
 - Semikolon ; indikerer ny rad
 - Eksempel: [1, 2, 3; 4 5 6]
- Karakterstrenger
 - Tegnene må angis innenfor to apostrofer ''
 - Eksempel: 'Dette er en streng'
- Avbrytes med CTRL + C

```
1 - z = input('Tast inn en verdi: ')
```

Command Window

```
>> kap7
```

```
Tast inn en verdi: [1 2 3; 4 5 6]
```

En matrise
tastes inn

```
z =
```

```
1     2     3
```

```
4     5     6
```

```
>> kap7
```

```
Tast inn en verdi: 'MATLAB er tøft'
```

Et sett
med tegn
tastes inn

```
z =
```

```
MATLAB er tøft
```

```
fx >>
```

Name ^	Value	Class
z	'MATLAB er ...	char

Editor - C:\Users\2900724\Dropbox\UiS\ING1x0\Forelesninger\2015\m-filer\kap7b.m

kap7b.m x +

```
1 - z = input('Tast inn et navn: ', 's')
```

Workspace

Name ^	Value	Class
z	'Tom'	char

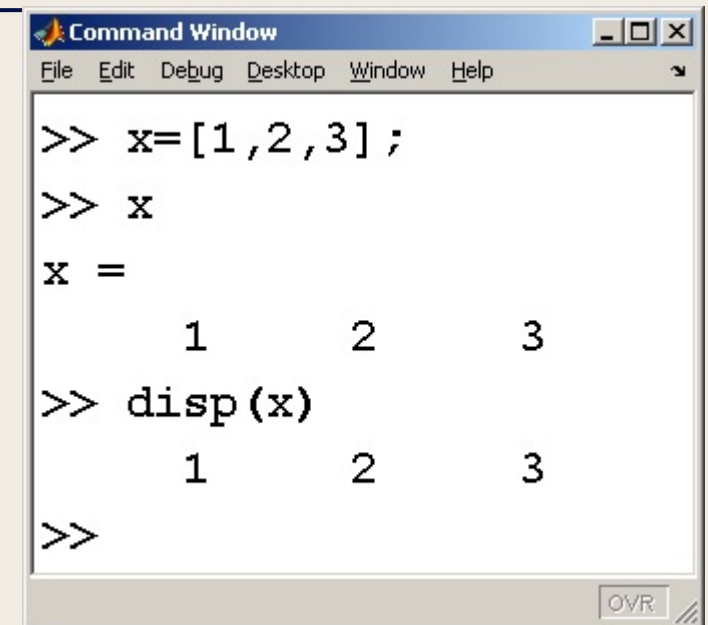
Indikerer at det som tastes inn skal tolkes som tekst

Command Window

```
>> kap7b  
Tast inn et navn: Tom  
  
z =  
  
Tom  
  
fx >>
```

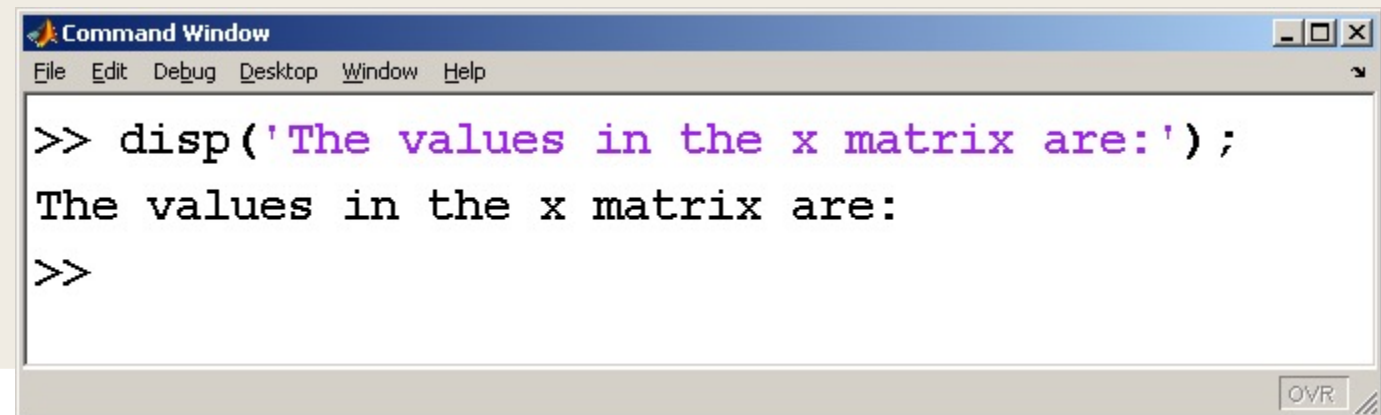
Utskrift (output): disp-funksjonen

- **disp**-funksjonen kan brukes til å gi ut tekst/verdier uten å skrive ut variabelnavnet, eller til å skrive ut en tekststreng



A screenshot of the MATLAB Command Window. The window has a title bar 'Command Window' and a menu bar with 'File', 'Edit', 'Debug', 'Desktop', 'Window', and 'Help'. The command history shows: `>> x=[1,2,3];`, `>> x`, and the output `x =` followed by a row vector `1 2 3`. Below that, the command `>> disp(x)` is entered, followed by the same output `1 2 3`. The prompt `>>` is shown at the bottom. An 'OVR' button is visible in the bottom right corner.

```
>> x=[1,2,3];
>> x
x =
     1     2     3
>> disp(x)
     1     2     3
>>
```



A screenshot of the MATLAB Command Window. The window has a title bar 'Command Window' and a menu bar with 'File', 'Edit', 'Debug', 'Desktop', 'Window', and 'Help'. The command history shows: `>> disp('The values in the x matrix are:');`, followed by the output `The values in the x matrix are:`. The prompt `>>` is shown at the bottom. An 'OVR' button is visible in the bottom right corner.

```
>> disp('The values in the x matrix are:');
The values in the x matrix are:
>>
```


En tekststreng er egentlig en vektor av tegn



The image shows a MATLAB environment with the following components:

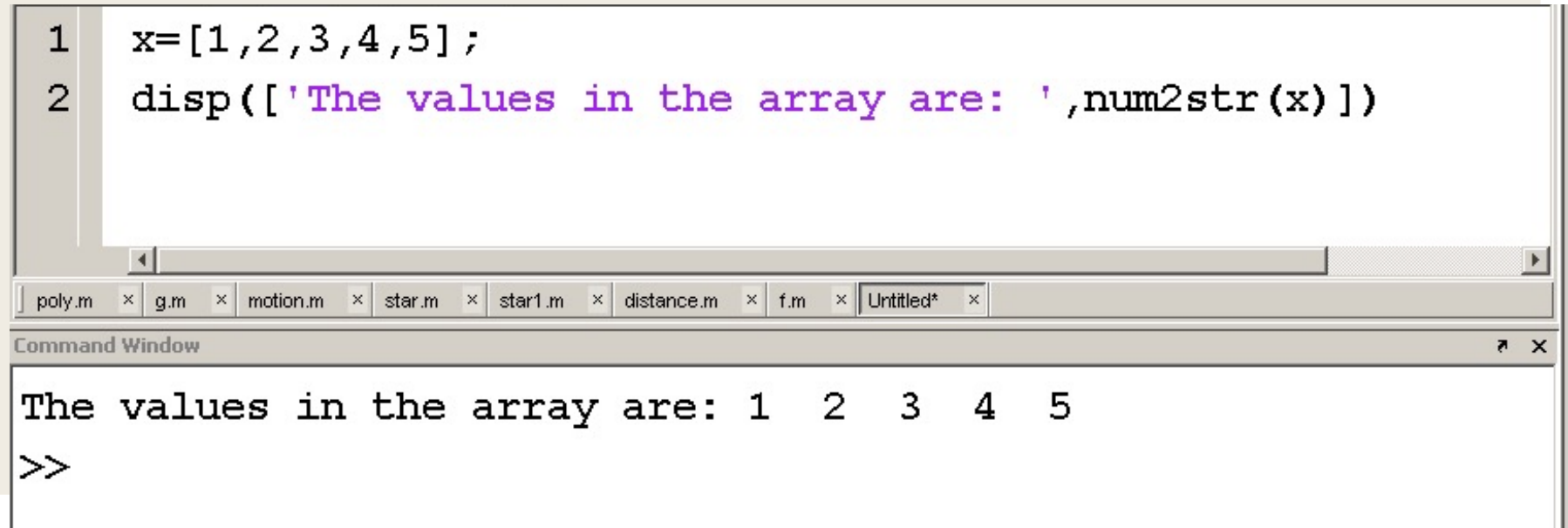
- Editor:** Contains the code `z = input('Tast inn et navn: ', 's')` on line 1.
- Command Window:** Shows the execution of `kap7b`, the prompt `Tast inn et navn: Tom`, and the assignment `z = Tom`.
- Workspace:** A table showing the variable `z` with the following properties:

Name	Value	Size	Class
z	'Tom'	1x3	char

A red oval highlights the workspace table, and an arrow points from a yellow box labeled "En tegntabell" to it.

Sette sammen tekst og tall

- Når både tekst og tallverdier skal skrives ut, må tallverdier konverteres til karakterer (tekst)
- Flere strenger kan settes sammen («concatenation»). Legg merke til bruken av [] innenfor ()
- **num2str(x)**-funksjonen gjør om *tall til streng*



The image shows a MATLAB script editor window with two lines of code. The first line defines an array `x` with values 1, 2, 3, 4, and 5. The second line uses the `disp` function to display a string concatenated with the array `x` converted to a string using `num2str(x)`. Below the script editor is the Command Window, which shows the output of the script: "The values in the array are: 1 2 3 4 5". The Command Window also shows the prompt `>>`.

```
1 x=[1,2,3,4,5];  
2 disp(['The values in the array are: ',num2str(x)])
```

The Command Window displays the output:

```
The values in the array are: 1 2 3 4 5  
>>
```

Hente og skrive data til fil

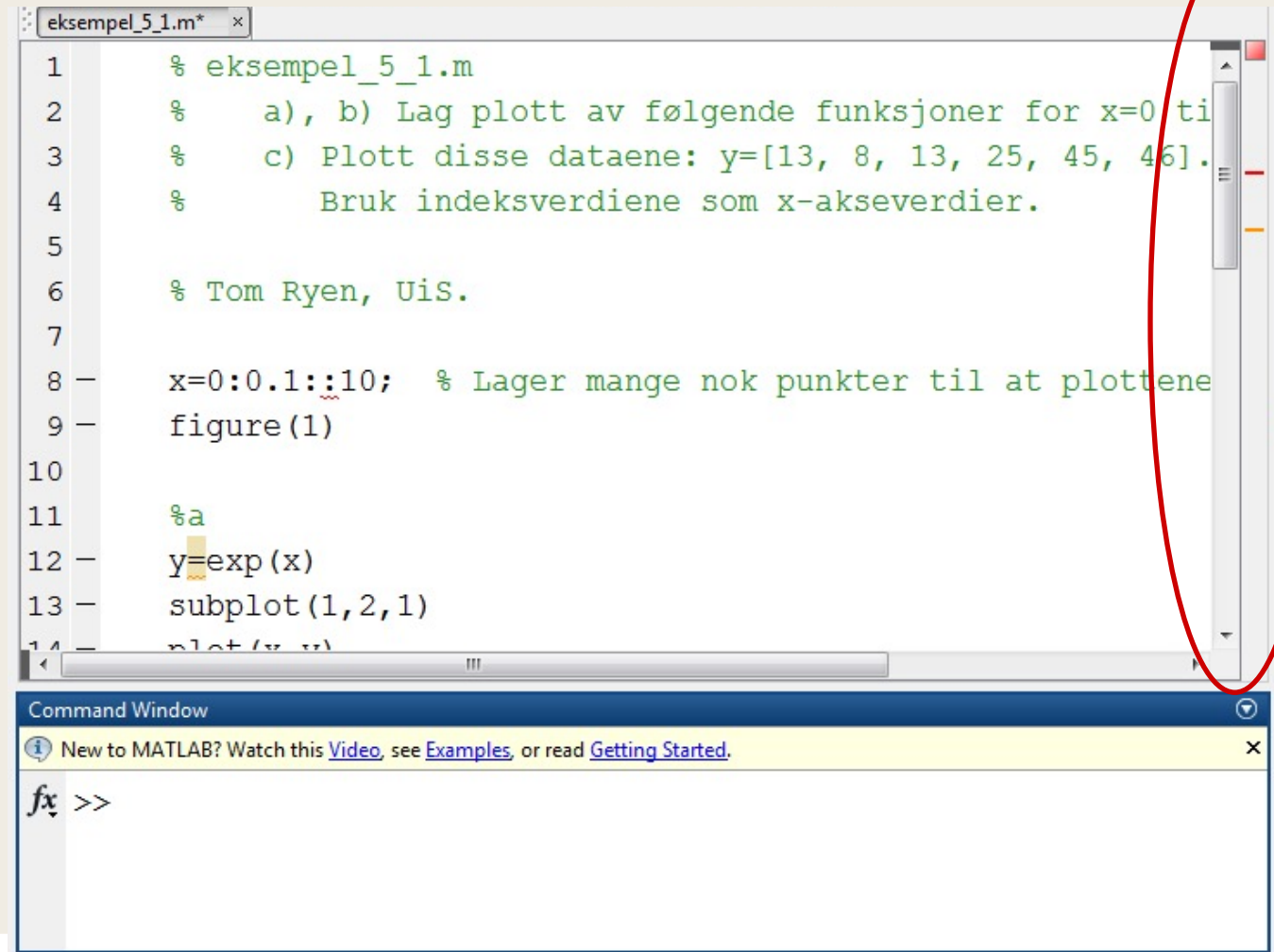
- Filformater
 - .mat MATLAB-datafil som inneholder variabler
 - **load**-funksjonen henter fra .mat-fil og inn i 'workspace'
 - **save**-funksjonen lagrer variabler i 'workspace' til .mat-fil
 - load og save kan også hente fra/ lagre til tekstfil (ASCII-fil.)
- Andre kjente datatyper
 - .xls, .xlsx Excel-regneark . Bruk **xlsread** og **xlswrite**
 - .jpg JPEG bildefil. Bruk **imread** og **imwrite**
 - ...
- se mer:
- >> **help fileformats**

Feilsøking i MATLAB-kode

- Ofte er det feil i koden i .m-filen vi har laget, og vi trenger måter å søke etter disse
- Dette kalles **feilsøking**; «debugging» på engelsk
- Det finnes god hjelpefunksjonalitet for feilsøking i MATLAB

«Error bar» til høyre i editorvinduet

- Linjer med feil markeres med rødt
- Linjer markert med gult har advarsler (som ofte kan ignoreres)



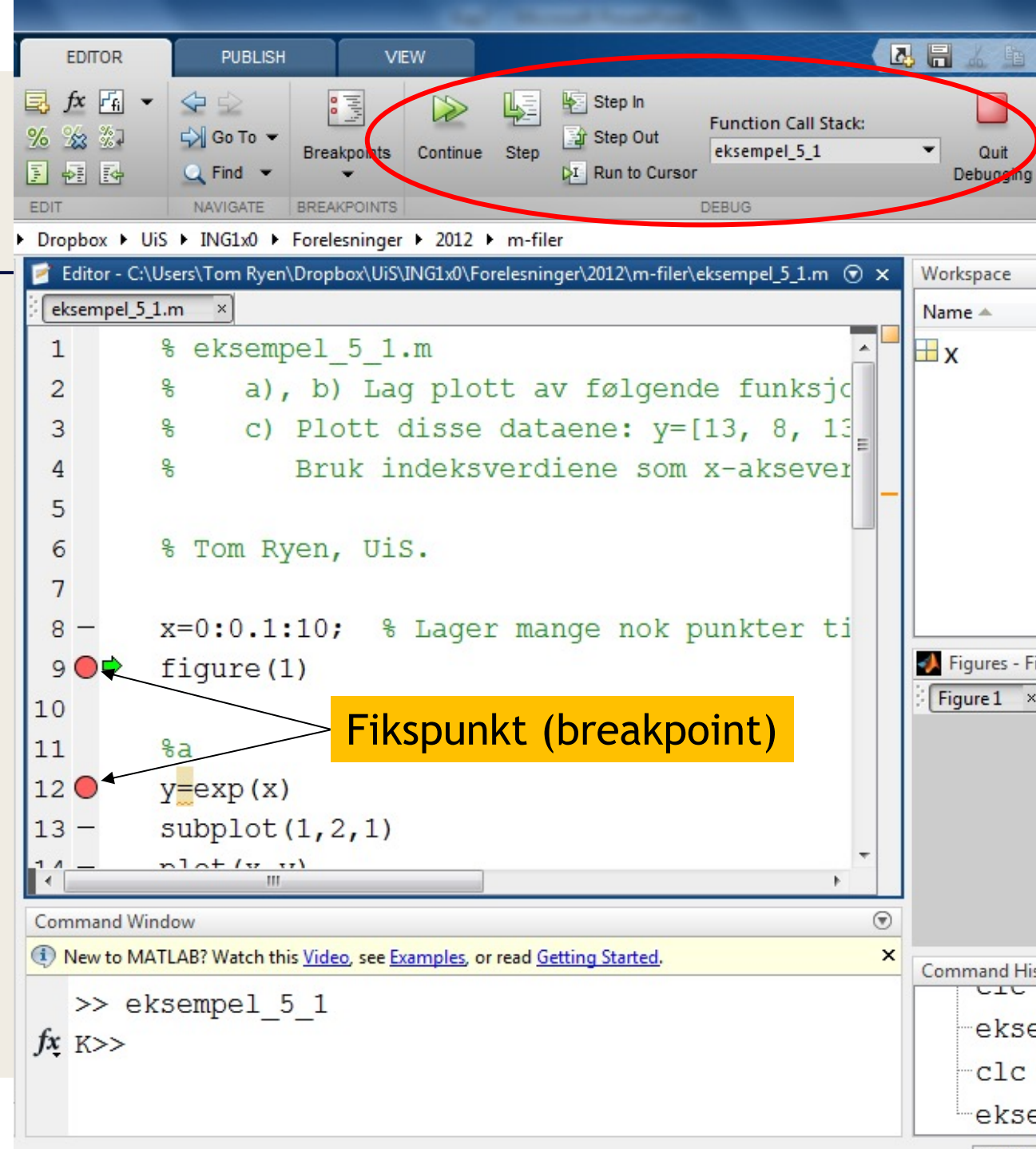
The image shows a MATLAB editor window titled 'eksempel_5_1.m'. The script contains the following code:

```
1 % eksempel_5_1.m
2 % a), b) Lag plott av følgende funksjoner for x=0 ti
3 % c) Plott disse dataene: y=[13, 8, 13, 25, 45, 46].
4 % Bruk indeksverdiene som x-akseverdier.
5
6 % Tom Ryen, UiS.
7
8 x=0:0.1::10; % Lager mange nok punkter til at plottene
9 figure(1)
10
11 %a
12 y=exp(x)
13 subplot(1,2,1)
14 plot(x,y)
```

The code is displayed with line numbers on the left. Lines 12 and 13 have yellow markers in the left margin, indicating warnings. Lines 8 and 9 have red markers in the left margin, indicating errors. A red circle is drawn around the right side of the editor window, highlighting the 'Error bar' area. Below the editor window is the 'Command Window' with a message: 'New to MATLAB? Watch this [Video](#), see [Examples](#), or read [Getting Started](#).' and a prompt 'fx >>'.

Breakpoint - «pauser» i programmet

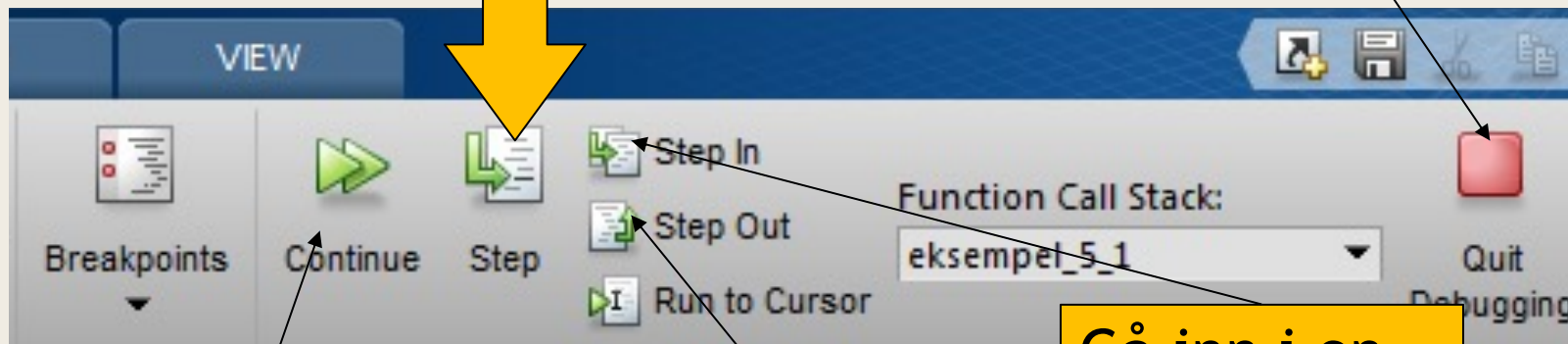
- VELDIG nyttig ved feilsøking!
- I kommando-vinduet vises **K>>** når programmet ennå ikke er ferdigkjørt.



Eksempel

Utfør en kodelinje om gangen

Avslutt debugginggen



Gå til neste breakpoint (eller ut programmet)

Gå ut av en (eventuell) funksjon

Gå inn i en (eventuell) funksjon

Tips for debugging

- Dersom du først skriver inn kommandoen:

```
>>dbstop if error
```

i command window, vil du automatisk havne inn i debug-modus dersom MATLAB finner en feil under kjøring