

---

# Matematikk 1000

## Øvingsoppgaver i numerikk – leksjon 4

### Løsningsforslag

---

#### Oppgave 1 – Funksjonsfiler

- b) Kommandoen `> help FunksjonenMin` gjør at dette blir skrevet til skjerm:

```
Funksjonen f(x)=sin(2x) - x^2.  
Funksjonen tar bare skalarer som input.
```

– altså det vi selv har kommentert i begynnelsen av funksjonsfila.

- c) Ved å endre den siste linja til

```
F=sin(2*x)-x.^2;
```

blir funksjonen i stand til å ta vektor-argument<sup>1</sup>. Vi plotter:

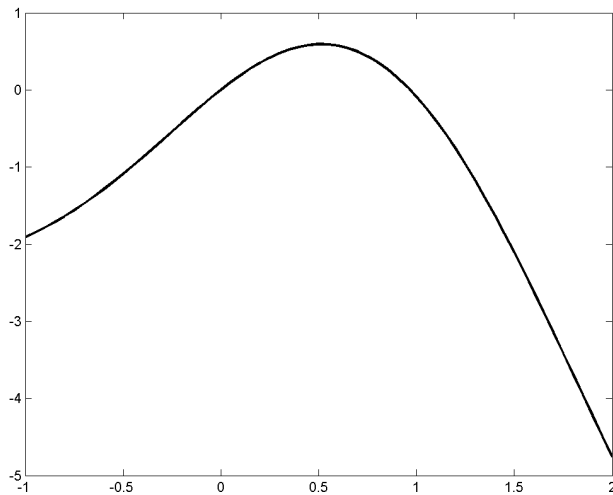
```
>> x=-1:.05:2;  
>> y=FunksjonenMin(x);  
>> plot(x,y,'k-', 'linewidth',2)
```

Resultatet kan sees i figur 1. Vi kunne gjort det hele litt mer kompakt også:

```
>> x=-1:.05:2;  
>> plot(x,FunksjonenMin(x),'k-', 'linewidth',2)
```

---

<sup>1</sup>Merk at den linja som sier at fila bare tar skalare argument bør oppdateres nå.



Figur 1: Plott av funksjonen  $f(x) = \sin(2x) - x^2$ .

## Oppgave 2 – Delt forskrift – if-satser

a) Slik kan vi leke litt med logiske uttrykk:

```
>> 1>0
ans = 1
```

Når vi skriver at  $1 > 0$ , som opplagt er sant, får vi 1 til svar. Her er altså 1 å tolke som 'sant'. Tilsvarende vil 0 bety 'usant' i denne sammenhengen:

```
>> 1<0
ans = 0
>> 2==2
ans = 1
>> 2==3
ans = 0
>> 2~=3
ans = 1
```

Vi kan også sette opp mer sammensatte logiske uttrykk:

```
> x=2;
> x>1 & x<3
ans = 1
> x>1 & x<1.5
ans = 0
> x>1 | x<1.5
ans = 1
```

I følge MATLAB er det altså riktig at, hvis  $x = 2$ , så er  $x > 1$  og  $x < 3$ . Derimot vil det være feil å si at  $x > 1$  og  $x < 1.5$ . Det er derimot riktig å si at  $x > 1$  *eller*  $x < 1.5$ .

b) Vi regner ut noen funksjonsverdier for DeltForskrift:

```
> DeltForskrift(2)
ans = 2
> DeltForskrift(1.999)
ans = 3.0000
> DeltForskrift(0)
ans = 3
> DeltForskrift(4)
ans = 14
```

Denne funksjonfila gir funksjonen

$$f(x) = \begin{cases} \cos(\pi x) + 2, & x < 2 \\ x^2 - 2, & x \geq 2 \end{cases} .$$

c) Om vi skriver

```
>> x=1:10;
>> x>5
```

får vi ut en vektor med 0 for alle elementer i  $x$  som er mindre eller lik 5 og 1 for alle elementer i  $x$  som er større enn 5:

```
ans =
     0     0     0     0     0     1     1     1     1     1
```

Dette kan vi utnytte:

```
(x<2).*(cos(pi*x)+2)
```

vil gi en vektor der alle elementer i  $(\cos(\pi x) + 2)$  ganges med null dersom  $x$ -elementet er mindre enn null og med én hvis ikke. Om vi legger til  $(x \geq 2) \cdot (x^2 - 2)$ , har vi fått med hele funksjonen:

```
1 function F=DeltForskriftV2(x)
2
3 % Funksjonen f(x) der x=cos(pi*x) + 2 for x<2
4 % og x^2-2 for x>=2
5 % Funksjonen kan ta vektor-argument.
6
7 F=(x<2).*(cos(pi*x)+2) + (x>=2).*(x.^2-2);
```

### Oppgave 3 – Hva gjør disse skriptene?

a) Her kjenner vi nok igjen  $abc$ -formelen for løsning av andregradslikninger:

$$ax^2 + bx + c = 0 \Leftrightarrow x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} .$$

Vi kjører skriptet for noen verdier av  $a$ ,  $b$  og  $c$ <sup>2</sup>:

```
>> format compact
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: -2
Gi verdien for c: -4
x1 =
    -1.2361
x2 =
     3.2361
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: -11
Gi verdien for c: 30
x1 =
     5
x2 =
     6
```

For noen verdier går det ikke så bra:

```
>> ABCformel
Gi verdien for a: 0
Gi verdien for b: 1
Gi verdien for c: 2
x1 =
    -Inf
x2 =
     NaN
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: 1
Gi verdien for c: 2
x1 =
    -0.5000 - 1.3229i
x2 =
    -0.5000 + 1.3229i
```

---

<sup>2</sup>» Format compact'-kommandoen gjør at MATLAB skriver ting med litt færre mellomrom.

I det siste eksemplet har vi fått komplekse løsninger. Vi minner om at “svarene” Inf og NaN står for uendelig (*infinity*) og udefinert (*not a number*). For andre verdier av  $a$ ,  $b$  og  $c$  får vi bare én løsning:

```
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: -6
Gi verdien for c: 9
x1 =
     3
x2 =
     3
```

- b) Dette skriptet undersøker om det aktuelle tallet,  $x$ , er positivt eller negativt. I tillegg undersøker det om  $x$  er et heltall. Dersom det *er* det, vil det også undersøke om det er et partall eller et oddetall. Vi viser noen eksempl:

```
>> TypeTall
Gi et tall 5
ans =
Tallet er positivt
ans =
Tallet er et oddetall
>> TypeTall
Gi et tall -8
ans =
Tallet er negativt
ans =
Tallet er et partall
>> TypeTall
Gi et tall pi
ans =
Tallet er positivt
ans =
Tallet er ikke et heltall
```

Om man vil unngå å få “ans=” i kommandovinduet, kan man bruke kommandoen “disp”; linje 4 endres til

```
disp('Tallet er negativt')
```

og man gjør tilsvarende i linje 8, 11 og 13 også.

- d) En kommentert versjon av TypeTall-skriptet, kan se slik ut:

```
1 % Skript som, for et gitt tall, avgjør om tallet er positivt
2 % eller negativt, om det er et heltall eller ikke, og - dersom
```

```

3 % det er et heltall - om det er et partall eller et oddetall.
4
5 % Leser inn tallet fra kommandovinduet
6 x=input('Gi et tall ');
7
8 % Avgjør om tallet er positivt eller negativt
9 if x<0
10     'Tallet er negativt'
11 else
12     'Tallet er positivt'
13 end
14
15 % Avgjør om tallet er et heltall
16 if round(x)~=x
17     'Tallet er ikke et heltall'
18 else
19     if round(x/2)==x/2           % Avgjør om heltallet er et partall
20         'Tallet er et partall'
21     else
22         'Tallet er et oddetall'
23     end
24 end

```

c) En litt mer “idiotsikker” versjon av skriptet ABCformel kan se slik ut:

```

1 % Skript som løser likninga  $a x^2 + b x + c = 0$ 
2
3 % Gir verdiene på a, b og c:
4 a=input('Gi verdien for a: ');
5 b=input('Gi verdien for b: ');
6 c=input('Gi verdien for c: ');
7
8 % Kontrollerer at det faktisk er en andregradsligning
9 if a==0
10     x=-c/b
11     return
12 end
13
14 % Undersøker om vi har to, én eller ingen reelle løsninger
15 if b^2-4*a*c<0           % Ingen reell løsning
16     'Ingen reell løsning'
17 elseif b^2-4*a*c==0     % En løsning
18     x=-b/(2*a)
19 else                     % To løsninger
20     x1=(-b-sqrt(b^2-4*a*c))/(2*a)
21     x2=(-b+sqrt(b^2-4*a*c))/(2*a)
22 end

```

Om vi ser på de samme eksemplene som i a) en gang til, får vi:

```
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: -11
Gi verdien for c: 30
x1 =
    5
x2 =
    6
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: 1
Gi verdien for c: 2
ans =
Ingen reell løsning
>> ABCformel
Gi verdien for a: 1
Gi verdien for b: -6
Gi verdien for c: 9
x =
    3
```

Her blir svaret 3 bare skrevet én gang. I tillegg er skriptet nå i stand til å håndtere førstegradsligninger også ( $a = 0$ ):

```
>> ABCformel
Gi verdien for a: 0
Gi verdien for b: -5
Gi verdien for c: 10
x =
    2
```