

UNIVERSITETET I STAVANGER

DET TEKNISK-NATURVITENSKAPELIGE FAKULTET

EKSAMEN I EMNENE:

BID260 Matlab for ingeniører
ING100 Ingeniørfaglig innføringsemne – Data og elektro
ING110 Ingeniørfaglig innføringsemne – Bygg
ING120 Ingeniørfaglig innføringsemne – Maskin
ING130 Ingeniørfaglig innføringsemne – Petroleum
ING140 Ingeniørfaglig innføringsemne – Kjemi

DATO: Torsdag 22. november 2012

VARIGHET: 3 timer

TILLATTE HJELPEMIDLER: Bestemt enkel kalkulator.

OPPGAVESETTET BESTÅR AV: 40 oppgaver på 16 sider + 4 siders vedlegg

MERK:

Alle oppgaver er av type «multiple choice» og alle oppgaver teller likt. Hver oppgave har bare ett riktig svar blant svaralternativene A, B, C og D. Hvert riktig svar gir 1 poeng. Feil svar eller flere svar på en oppgave gir 0 poeng. Du skal besvare ved å fylle inn i et eget skjema.

Bruk blå eller svart penn, og marker ditt valg ved å sette en strek (og ikke et kryss) for ditt valg. For eksempel slik:

Oppgave

	A	B	C	D
1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Dersom du har fylt inn feil, kan du rette feilen ved å sette et solid kryss over valget du forkaster. Eksempel:

Oppgave

	A	B	C	D
1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Du kan få utdelt et nytt skjema for utfylling dersom du trenger det, men bare ett skjema skal leveres inn til slutt. Dersom du ønsker å ta vare på dine egne svar, så skriv de ned på et eget kladdeark og ta dette med hjem.

LES OPPGAVENE NØYE!

Oppgave 1-10: Hva blir skrevet i kommandovinduet når denne Matlab-koden kjøres:

Oppgave 1 $x = 4 * 5;$ $y = x$			
A $y =$ x	B $y =$ 20	C $x =$ $4 * 5$	D $x =$ 20

Oppgave 2 $x = 4^3;$ $y = x + 2;$ $x = y$			
A $x =$ 14	B $x =$ 64 $y =$ 66 $x =$ 64	C $x =$ y	D $x =$ 66

Oppgave 3 $a = 1:5;$ $b = a * 3$			
A $b =$ 15	B $b =$ 3 2 3 4 5		
C $b =$ 3 6 9 12 15	D $b =$ 3:15		

Oppgave 4 $a = [1, 2, 3; 4, 5, 6];$ $b = a'$			
A $b =$ 1 2 3 4 5 6	B $b =$ 1 2 3 4 5 6	C $b =$ 1 4 2 5 3 6	D $b =$ a'

Oppgave 5

```
c = [1,2,3;4,5,6];
d = [c; 2 2 2]
```

A	B	C	D
d =	d =	d =	d =
1 4 2	1 2 3 4 5 6	2 2 2	1 2 3
2 5 2	2 2 2	2 2 2	4 5 6
3 6 2		2 2 2	2 2 2

Oppgave 6

```
A = [1,2,3,4; 4,5,6,7; 1,2,3,4];
B = A(2:end,3:4)
```

A	B	C	D
B =	B =	B =	B =
5 6 7	6 7	3 4	2 3
2 3 4	3 4	6 7	3 4
		3 4	

Oppgave 7

```
x = uint8(8);
y = uint8(8);
if x ~= y
    disp('november')
else
    disp('desember')
end
```

A	B	C	D
(ingen ting skrives ut)	november desember	november	desember

Oppgave 8

```
j = 3;
for i=1:6
    if i > j
        disp(i)
    end
end
```

A	B	C	D
3	1	1 2 3 4 5 6	4
	2		5
	3		6

Oppgave 9

```
n = 3;
while n >= 1
    disp(n)
    n=n-2;
end
```

A	B	C	D
3	3	1	3
	1	-1	3 - 2

Oppgave 10

```
A = ones(3,2);
for i=1:3
    for j=4:5
        A(i,j-3) = i*j;
    end
end
A
```

A	B	C	D
A =	A =	A =	A =
1 2 3	1 1	4 5	1 2
4 5 5	1 1	8 10	2 4
	1 1	12 15	3 6

Oppgave 11 - 20: Hva er den rette koden for å få utført oppgaven?

Oppgave 11

Lag et program som lar noen taste inn et heltall i tallområdet 1 til 6. Så skal Matlab finne et tilfeldig tall i dette tallområdet. Dersom tallet er det samme som inntastet tall, skal teksten "Gratulerer, du vant!" komme i kommandovindu. Hvis ikke, skal "Beklager, du tapte." komme fram.

A

```
inntall = input('Tast inn et tall mellom 1 og 6: ');
tilftall = randi(6);
disp(['Matlab sitt tall er ',num2str(tilftall)])
if (inntall == tilftall)
    disp('Gratulerer, du vant!')
else
    disp('Beklager, du tapte.')
end
```

B

```
disp('Tast inn et tall mellom 1 og 6: ');
tilftall = inntall;
disp(['Matlab sitt tall er ',num2str(tilftall)])
if (inntall == tilftall)
    disp('Gratulerer, du vant!')
else
    disp('Beklager, du tapte.')
end
```

C

```
inntall = input('Tast inn et tall mellom 1 og 6: ');
tilftall = 1:6;
disp(['Matlab sitt tall er ',num2str(tilftall)])
if (inntall ~= tilftall)
    disp('Gratulerer, du vant!')
else
    disp('Beklager, du tapte.')
end
```

D

```
inntall = input('Tast inn et tall mellom 1 og 6: ');
tilftall = randi(6);
disp(['Matlab sitt tall er ',num2str(tilftall)])
while (inntall == tilftall)
    disp('Gratulerer, du vant!')
end
disp('Beklager, du tapte.')
```

Oppgave 12

Lag et program som lager en 4x4-matrise med tilfeldige tall. Alle tallene skal være mellom 0 og 1. Så skal programmet finne det største tallet i hele matrisen.

A

```
A = rand(4)
max(max(A))
```

B

```
A = rand(4x4)
max(A)
```

C

```
A = randi(4)
max(A)
```

D

```
A = randi(4,0-1)
max(A)
```

Oppgave 13

Lag et program som lager en 4x4-matrise med tilfeldige tall. Tallene skal være uniformt fordelt i tallområdet [-1,1]. Så skal programmet finne antall negative tall i matrisen.

A

```
B = rand(4)
length(find(B<0))
```

B

```
B = rand(4,-1,1)
length(find(B<0))
```

C

```
B = 2*rand(4)-1
length(find(B<0))
```

D

```
B = rand(4)-1
length(find(neg(B)))
```

Oppgave 14

Den generelle likningen for distansen (d) som et legeme i fritt fall har tilbakelagt fra tiden $t_0=0$ (sett vekk fra luftmotstanden) er gitt av

$$d = \frac{gt^2}{2}$$

der t er tiden. Anta at tyngdeakselerasjonen $g = 9,8 \text{ m/s}^2$. Lag en tabell som viser tiden versus distansen for hvert tiende sekund fra 0 til 100 sekunder. Tabellen skal ha to kolonner.

A g=9,8; t=0:100; d=1/2*g*t^2; tab=[t,d]	B t=0:10:100; d=1/2*g*t^2; tab=[t,d]
C d=1/2*g*t.^2; g=9.8; t=0:100; tab=[t,d]	D g=9.8; t=0:10:100; d=1/2*g*t.^2; tab=[t',d']

Oppgave 15

Lag en 3x3 matrise med tilfeldige heltall i tallområdet [1,5]. Finn ut om summen av første og andre rad er lik. Det skal komme en melding om dette i kommandovinduet.

A

```
F = randi(3x3)
s1 = sum(F(1,:));
s2 = sum(F(2,:));
if s1 = s2
    disp('Summen av første og andre rad er lik.')
end
```

B

```
F = randi(3,3)
s1 = sum(F(:,1));
s2 = sum(F(:,2));
if s1 == s2
    disp('Summen av første og andre rad er lik.')
end
```

C

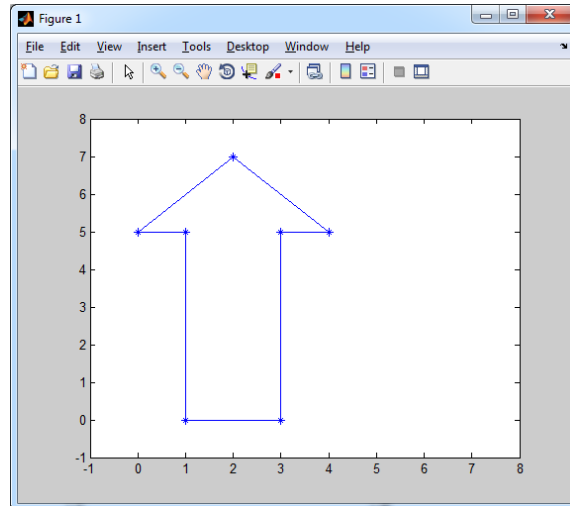
```
F = rand(3)
s1 = sum(F(1));
s2 = sum(F(2));
if s1 == s2
    disp('Summen av første og andre rad er lik.')
end
```

D

```
F = randi(5,3)
s1 = sum(F(1,:));
s2 = sum(F(2,:));
if s1 == s2
    disp('Summen av første og andre rad er lik.')
end
```

Oppgave 16

Lag følgende plott:



A

```
x = [ 1 3 3 4 2 0 1 1 ];  
y = [ 0 0 5 5 7 5 5 0 ];  
plot(x,y,'*-')  
axis([-1 8 -1 8])
```

B

```
x = [ 1 3 3 4 2 0 1 1 ];  
y = sin(x);  
plot(x,y,'*')  
axis([-1 8 -1 8])
```

C

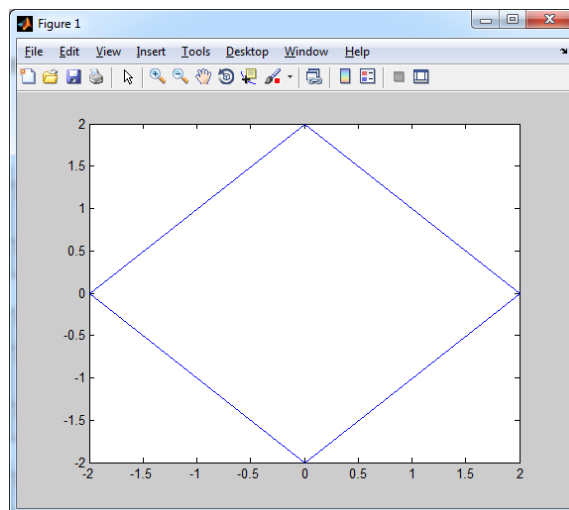
```
[x,y] = arrow(3,7);  
plot(x,y,':d')  
axis([-1 8 -1 8])
```

D

```
[x,y] = arrow(3,7);  
plot(x,y,'*-')  
axis([-1 8 -1 8])
```

Oppgave 17

Lag følgende plott:



A

```
x = [ 2 2 0 0];  
y = [ 0 0 2 2];  
plot(x,y)
```

B

```
x = [ 0 2 0 -2];  
y = [ 0 0 -2 2];  
plot(x,y)
```

C

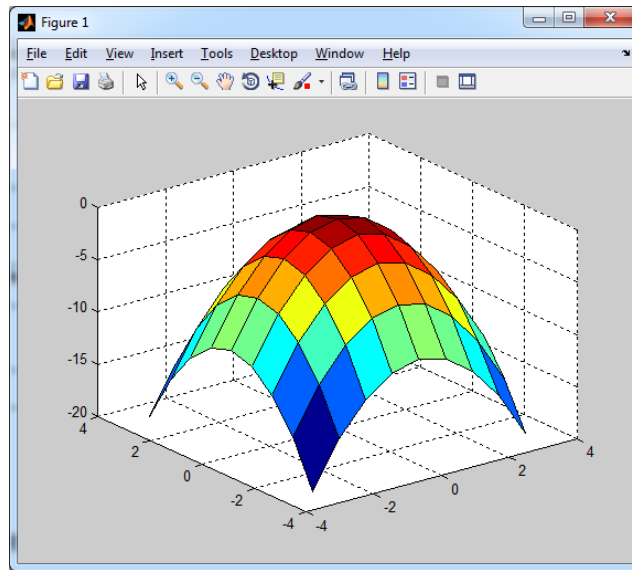
```
r = 2;  
v = 0:90:360;  
for k=1:length(v)  
    x(k) = r*cosd(v(k));  
    y(k) = r*sind(v(k));  
end  
plot(x,y)
```

D

```
r = 2;  
v = 0:90:360;  
for k=1:length(v)  
    x = r*cos(v);  
    y = r*sin(v);  
end  
plot(x,y)
```

Oppgave 18

Lag følgende plott:



(I Matlab vil figuren ha farger.)

A

```
x = -pi:pi/4:pi;
y = x;
[X,Y] = meshgrid(x,y);
surf(X,Y,X.^2 + Y.^2)
```

B

```
x = -pi:pi/4:pi;
y = x;
[X,Y] = meshgrid(x,y);
Z = -(X.^2 + Y.^2);
surf(X,Y,Z)
```

C

```
x = -pi:pi/4:pi;
y = x;
[x,y] = meshgrid(X,Y);
surf(X,Y,X.^2 + Y.^2)
```

D

```
x = 0:pi/4:pi;
y = x;
[X,Y] = surfgrid(x,y);
Z = -(X.^2 + Y.^2);
smurf(X,Y,Z)
```

Oppgave 19

Finn arealet (A) av en trekant basert på flere verdier av grunnlinje (g) og høyde (h):

$$A = \frac{1}{2} \cdot g \cdot h$$

La grunnlinjen ha verdiene 3, 4, ..., 12. La høyden ha verdiene 2, 3, ..., 6. Altså: Finn 50 svar for A.

A

```
g = 3:12;
h = 2:6;
[G,H]=meshgrid(g,h);
R = 0.5*G.*H
```

B

```
g = 3:12;
h = 2:6;
[g,h]=meshgrid(G,H);
R = 0.5*G*H
```

C

```
g = 3:12;
h = 2:6;
R = 0.5*g.*h
```

D

```
g = 3,4,...,12;
h = 2,3,...,6;
[g,h]=meshgrid(G,H);
R = 0.5*G.*H
```


Oppgave 20

Lag en funksjon med navn `sluttverdi`. Funksjonen skal ha tre innparametere og en utverdi. Funksjonen skal finne sluttverdien til et bankinnskudd ut fra følgende formel:

$$SV = NV(1 + i)^n$$

der SV er sluttverdi, NV er nåverdi, i er innskuddsrente (som kommatall) og n er antall år.

A

```
function SV=sluttverdi(NV)
NV.*(1+i).^n = SV;
```

B

```
function SV=sluttverdi(NV, i, n)
SV = NV.*(1+i).^n;
```

C

```
function sluttverdi=SV(NV, i, n)
SV = NV(1+i)^n;
```

D

```
function [NV,i,n]=sluttverdi(SV)
SV = NV.*(1+i).^n;
```

Oppgave 21 - 25: Disse programmene får feilmelding når de kjøres. Hva er grunnen til feilmeldingen?

Oppgave 21

```
x = [0 2 4];
x = y;
```

Feilmelding:

```
??? Undefined function or variable 'y'.
```

A

Man prøver å sette x lik y , men y finnes ikke, verken som funksjon eller variabel.

B

Man har semikolon i begge setninger. Da får vi en feilmelding fordi ingen ting blir skrevet ut.

C

Variabelen y skulle hatt apostrofer, det vil si at andre linje skulle vært slik: `x = 'y'`;

D

Første element i x er lik null. Det gjør at y blir udefinert.

Oppgave 22

```
x=1:10;
y=sin(x);
Plot(x,y)
```

Feilmelding:

```
??? Undefined function or method 'Plot' for input arguments of type 'double'.
```

A

Man prøver å plote en sinuskurve, men x og y er av datatypen 'double', og det går ikke an.

B

Man putter en kolon i x , som videre brukes som parameter i en sinus-funksjon. Det er årsaken til feilmeldingen.

C

Man har glemt semikolon på tredje linje. Det er grunnen til feilmeldingen.

D

Matlab-funksjonen `Plot` finnes ikke. I hvert fall ikke i aktuell katalog eller tilgjengelige kataloger.

Oppgave 23

```
x = [0 2 4];
y = x;
for k=1:3
    y(x) = 4;
end
```

Feilmelding:

```
??? Subscript indices must either be real positive integers or logicals.
```

A

For-løkken går bare 3 runder, men skulle gått 4, siden y(x) settes lik 4.

B

Indeksverdier kan ikke være lik 0 eller negative, men må være positive heltall. Første element i x er 0, og det forårsaker denne feilmeldingen.

C

Man kan ikke sette y lik 4 i alle elementer, når det ligger tall der fra før.

D

Siden variabelen k ikke brukes, blir alle subskriptindisiene ulogiske.

Oppgave 24

```
a = [3 4 5];
b = [6 7 8 9];
c = a + b
```

Feilmelding:

```
??? Error using ==> plus
Matrix dimensions must agree.
```

A

Man kan ikke addere to matriser.

B

Matrisene må være «enige», det vil si inneholde samme tallverdier.

C

Matrisene, som her er to vektorer, har ulik dimensjon. Man kan ikke legge sammen to matriser som ikke har samme dimensjon.

D

Her er det to vektorer og ikke to matriser. Det går ikke an å legge sammen to vektorer.

Oppgave 25

```
a = [ 4 5 6 7 ];
for k=1:7
    b = a(1:7);
end
```

Feilmelding:

```
??? Index exceeds matrix dimensions.
```

A

Vektoren a har bare 4 elementer. Her forsøkes det å hente ut element 1 til 7, som vil gå ut over dimensjonen på vektoren.

B

Indeksen eksisterer matrisedimensjonen, som betyr at man prøver å lage en matrise av en vektor.

C

Setningen inne i for-løkken må erstattes med denne setningen:
 $b = a(k);$
 Da går det greit.

D

Det skulle stått
 else
 i stedet for
 end
 helt til slutt.

Oppgave 26 - 28: Om datatyper.

<p>Oppgave 26 Hvilken av følgende kommandoer gir dette svaret i kommandovinduet?</p> <pre>ans = 104.6000</pre>			
A char(104.6)	B uint8(104.6)	C double(104.6)	D int64(104.6)

<p>Oppgave 27 Hva blir vist i kommandovindu når denne kommandoen kjøres?</p> <pre>uint16(3456.2)</pre>			
A ans = 255	B ans = 3456	C ans = 510	D ans = 3456.2

<p>Oppgave 28 Hva blir vist i kommandovindu når denne kommandoen kjøres?</p> <pre>x = [97 98 99]; disp(['x = [', num2str(x), ']'])</pre>			
A x = abc	B x = 'abc'	C x = [97 98 99]	D x = [a b c]

Oppgave 29 - 40: Forskjellige typer oppgaver.

<p>Oppgave 29 Hva blir skrevet ut i kommandovinduet? Anta at Symbolic Math Toolbox er installert.</p> <pre>syms x; expand((x-5)*(x+5))</pre>			
A ans = (x-5)*(x+5)	B ans = (x - 5) * (x + 5)	C ans = x^2 - 25	D ans = x^2 - 10*x - 25

Oppgave 30

Hva gjør dette Matlab-programmet? Anta at Symbolic Math Toolbox er installert.

```
y = sym('x^3')
yd = diff(y)
ydd = diff(y,2)
subplot(1,2,1)
ezplot(yd,[-2,2])
subplot(1,2,2)
ezplot(ydd,[-2,2])
```

A

Programmet finner differensiallikninger med en og to ukjente, og plottes disse med "easy plot"-funksjonen, som brukes til å gjøre plottingen enklere.

B

Programmet finner den deriverte og den integrerte av $y = x^3$, og plottes disse i to plott som står over hverandre.

C

Programmet finner den deriverte og den dobbeltderiverte av $y = x^3$, og plottes disse for intervallet $x=[-2,2]$. Plottene står ved siden av hverandre i en og samme figur.

D

Programmet tegner to differensierte kurver, der den ene plottes konvekst og den andre konkavt, begge for intervallet $x=[-2,2]$. Plottene står ved siden av hverandre.

Oppgave 31

Finn rett Matlab-kode for et program som lar en bruker få taste inn et månedsnummer og som gir et svar på hvilken årstid den måneden tilhører. Dersom noe annet enn tallene 1 til 12 tastes inn, skal det komme en melding om ugyldig inntasting.

A

```
maned = input('Tast inn månedsnummer: ');
if maned <= 12
    case {12,1,2}
        disp('Det er vinter')
    case {3,4,5}
        disp('Det er vår')
    case {6,7,8}
        disp('Det er sommer')
    case {9,10,11}
        disp('Det er høst')
    else
        disp('Ugyldig inntasting')
end
```

B

```
maned = input('Tast inn månedsnummer: ');
switch maned
    case {12,1,2}
        disp('Det er vinter')
    case {3,4,5}
        disp('Det er vår')
    case {6,7,8}
        disp('Det er sommer')
    case {9,10,11}
        disp('Det er høst')
    otherwise
        disp('Ugyldig inntasting')
end
```

C

```
disp('Tast inn månedsnummer: ');
switch maned <= 12
    case {12,1,2}
        disp('Det er vinter')
    case {3-5}
        disp('Det er vår')
    case {6-8}
        disp('Det er sommer')
    case {9-11}
        disp('Det er høst')
    otherwise
        disp('Ugyldig inntasting')
end
```

D

```
maned = input('Tast inn månedsnummer: ');
if maned == {12,1,2}
    disp('Det er vinter')
elseif maned == {3,4,5}
    disp('Det er vår')
elseif maned == {6,7,8}
    disp('Det er sommer')
elseif maned == {9,10,11}
    disp('Det er høst')
else
    disp('Ugyldig inntasting')
end
```

Oppgave 32

Hva gjør dette programmet?

```
a = randi(20,1);
b = randi(20,1);
while(1)
    svar = input([num2str(a), ' + ', num2str(b), ' = ']);
    if svar == (a+b)
        break;
    end
    disp('Feil. Prøv igjen:')
end
disp('Gratulerer, rett svar!')
```

A

Det setter opp en addisjon mellom to tilfeldige heltall fra 1 til 20. Det er brukeren av programmet som må svare. Dersom feil svar tastes inn, vil man få melding om det og man må prøve på nytt. Dette gjentas helt til man taster rett svar.

B

Programmet vil gå i en evig løkke, på grunn av at det står while(1). Samme hva brukeren gjør vil man ikke komme videre. Eneste utvei er å taste Ctrl + c når man står i kommandovindu med markøren.

C

Programmet setter opp en addisjon mellom to tilfeldige ASCII-tegn. Man må finne det nye ASCII-tegnet for å komme videre. Man vil fortsette og fortsette helt til man klarer det, og da vil "Gratulerer, rett svar!" bli skrevet ut.

D

Programmet gjør ingen ting, men vil bare forårsake en feilmelding på grunn av at det ikke går an å skrive while(1).

Oppgave 33

Hva er rett kode for et Matlab-program som finner en tilnærmet løsning av π med bruk av Leibniz formel, der 1, 2, ... n ledd i Leibniz-rekken er tatt med. Det lages et plott av rekken. Brukeren av programmet får velge n.

$$\text{Leibniz formel: } \pi = 4 \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1}$$

A

```
n = input('Tast inn grad av nøyaktighet: ');
minPi = 4;
for k=1:n
    denne = 4*(-1)^k / (2*k + 1);
    minPi = [minPi minPi(end)+denne];
end
minPi
plot(0:n,minPi)
```

B

```
n = disp('Tast inn grad av nøyaktighet: ');
for k=1:n
    denne = 4*(-1)^k / (2*k + 1);
    minPi = [minPi minPi(end)];
end
minPi
plot(0:n,minPi)
```

C

```
n = input('Tast inn grad av nøyaktighet: ');
for k=1:n
    minPi = 4*(-1)^k / (2*k + 1);
end
minPi
plot(0:n,minPi)
```

D

```
n = input('Tast inn grad av nøyaktighet: ');
minPi = 4;
for k=1:n
    minPi = sum( 4*(-1)^k / (2*k + 1) );
end
minPi
plot(0:n,minPi)
```

Oppgave 34

Hva blir skrevet i kommandovindu når denne koden kjøres og tallet 5 tastes inn?

```
n = input('Legg inn et positivt heltall mellom 1 og 10: ');
n = floor(n);
if ( n<1 || n>10 )
    disp('Ugyldig tall.')
else
    x = ones(1,n);
    i=2;
    while i<=n
        x(i) = x(i-1) + i;
        i = i + 1;
    end
    disp(x);
end
```

A 5	B 1 2 3 4 5
C 5 6 7 8 9	D 1 3 6 10 15

Oppgave 35

Med samme Matlab-kode som i oppgave 34: Hva blir skrevet i kommandovindu dersom tallet -3 tastes inn?

A Ugyldig tall.	B 1 2 3 4 5
C -3 -2 -1 0 1	D Ugyldig tall. 1 1 1

Oppgave 36 - 40:

Anta at filen f.m ligger i aktiv katalog og har dette innholdet:

```
function [z,w] = f(a,b,c)
    if nargin < 3
        c = 1;
    end
    if nargin < 2
        b = 1;
    end

    z = a + b + c;
    w = a*b*c;
end
```

Hva blir skrevet i kommandovindu når følgende kode kjøres:

Oppgave 36 <code>[z,w] = f(1,2,3)</code>			
A <code>z =</code> 3 <code>w =</code> 1	B <code>z =</code> <code>w =</code>	C <code>z =</code> 1 + 2 + 3 <code>w =</code> 1*2*3	D <code>z =</code> 6 <code>w =</code> 6

Oppgave 37 <code>f(7,7,7)</code>			
A <code>ans =</code>	B <code>ans =</code> 7	C <code>ans =</code> 21	D <code>ans =</code> 343

Oppgave 38 <code>a = [2 3 4];</code> <code>[x,y] = f(a(1), a(2), a(3))</code>			
A <code>z =</code> 6 <code>w =</code> 8	B <code>z =</code> 6 <code>w =</code> 6	C <code>x =</code> 4 <code>y =</code> 2	D <code>x =</code> 9 <code>y =</code> 24

Oppgave 39 <code>c = [4 5 6];</code> <code>for k=1:3</code> <code>if k>2</code> <code>[d,e] = f(c);</code> <code>end</code> <code>end</code> <code>d</code>			
A <code>d =</code> 6 7 8	B <code>d =</code> 15 <code>e =</code> 120	C <code>d =</code> 8 9 10	D (Ingen ting blir skrevet ut)

Oppgave 40 <code>g = [2 2; 2 2];</code> <code>h = [1 2; 3 4];</code> <code>f(g,h)</code>			
A <code>ans =</code> 8 12 8 12	B <code>ans =</code> 4 5 6 7	C <code>ans =</code> 4	D <code>ans =</code> 3 4 5 6

Matlab-kommandoer, Matlab-funksjoner og spesialtegn

man bør kjenne til i emnet BID260 Matlab for ingeniører / ING1x0 Ingeniørfaglig innføringsemne, felles del. (Dette kommer som vedlegg til eksamensoppgaven.)

Special Characters	Matrix Definition
[]	forms matrices
()	used in statements to group operations; used with a matrix name to identify specific elements
,	separates subscripts or matrix elements
;	separates rows in a matrix definition; suppresses output when used in commands
:	used to generate matrices; indicates all rows or all columns
%	comment
%%	cell divider
'	begins and ends a string

Commands	Basic Workspace Commands
help	opens the help function
ans	default variable name for results of MATLAB calculations
clc	clears command screen
clear	clears workspace
close	Close current figure window
close all	Closes all the open figure windows
diary	saves both commands issued in the workspace and the results to a file
date	returns the date
who	lists variables in memory
whos	lists variables and their sizes
intmax	returns the largest possible integer number used in MATLAB
intmin	returns the smallest possible integer number used in MATLAB
realmax	returns the largest possible floating-point number used in MATLAB
realmin	returns the smallest possible floating-point number used in MATLAB

Special Functions	Functions with Special Meaning That Do Not Require an Input
pi	numeric approximation of the value of π
i	imaginary number
Inf	Infinity
NaN	not a number

Functions	Elementary Math
abs	computes the absolute value of a real number or the magnitude of a complex number
exp	computes the value of e^x
factorial	calculates the factorial
isprime	determines whether a value is prime
isreal	determines whether a value is real or complex
log	computes the natural logarithm, or log base e (\log_e)
log10	computes the common logarithm, or log base 10 (\log_{10})
primes	finds the prime numbers less than the input value
prod	multiplies the values in an array
rem	calculates the remainder in a division problem
sqrt	calculates the square root of a number
sum	sums the values in an array

Functions	Trigonometry
sin	computes the sine, using radians as input
cos	computes the cosine, using radians as input
tan	computes the tangent, using radians as input
sind	computes the sine, using angles in degrees as input
cosd	computes the cosine, using angles in degrees as input
tand	computes the tangent, using angles in degrees as input

Functions	Random Numbers
rand	calculates evenly distributed random numbers
randn	calculates normally distributed (Gaussian) random numbers
randi	calculates evenly distributed random integers

Functions	Rounding
ceil	rounds to the nearest integer toward positive infinity
fix	rounds to the nearest integer toward zero
floor	rounds to the nearest integer toward minus infinity
round	rounds to the nearest integer

Functions	Data Analysis
length	determines the largest dimension of an array
size	determines the number of rows and columns in an array
sum	sums the values in an array
prod	multiplies the values in an array
sort	sorts the elements of a vector
sortrows	sorts the rows of a vector on the basis of the values in the first column
max	finds the maximum value in an array and determines which element stores the maximum value
min	finds the minimum value in an array and determines which element stores the minimum value
mean	computes the average of the elements in an array
median	finds the median of the elements in an array
mode	finds the most common number in an array
nchoosek	finds the number of possible combinations when a subgroup of k values is chosen from a group of n values

Functions	Matrix Formulation, Manipulation, and Analysis
meshgrid	maps vectors into a two-dimensional array
diag	extracts the diagonal from a matrix
fliplr	flips a matrix into its mirror image from left to right
flipud	flips a matrix vertically
linspace	linearly spaced vector function
logspace	logarithmically spaced vector function
isempty	true for empty array

Functions	Two-Dimensional Plots
plot	creates an x–y plot
plotyy	creates a plot with two y -axes
bar	generates a bar graph
hist	generates a histogram
semilogx	generates an x–y plot with the x -axis scaled logarithmically
semilogy	generates an x–y plot with the y -axis scaled logarithmically
loglog	generates an x–y plot with both axes scaled logarithmically
contour	generates a contour map of a three-dimensional surface

Functions	Three-Dimensional Plots
plot3	generates a three-dimensional line plot
mesh	generates a mesh plot of a surface
surf	generates a surface plot

Special Characters Control of Plot Appearance			
Indicator	Line Type	Indicator	Point Type
-	Solid	.	point
:	dotted	o	circle
-.	Dash-dot	x	x-mark
--	dashed	+	Plus
		*	Star
Indicator	Color	s	square
b	blue	d	diamond
g	green	^	triangle down
r	red	^	triangle up
c	cyan	<	triangle left
m	Magenta	>	triangle right
y	Yellow	p	pentagram
k	Black	h	hexagram

Functions	Figure Control and Annotation
figure	opens a new figure window
subplot	divides the graphics window up into sections available for plotting
hold on	instructs MATLAB not to erase figure contents before adding new information
hold off	instructs MATLAB to erase figure contents before adding new information
title	adds a title to a plot
xlabel	adds a label to the x -axis
ylabel	adds a label to the y -axis
zlabel	adds a label to the z -axis
legend	adds a legend to a graph
text	adds a text box to a graph
axis	freezes the current axis scaling for subsequent plots or specifies the axis dimensions
axis equal	forces the same scale spacing for each axis
grid	adds a grid to the current plot only
grid on	adds a grid to the current and all subsequent graphs in the current figure
grid off	turns the grid off

Functions	Function Creation and Use
function	identifies an M-file as a function
nargin	determines the number of input arguments in a function
nargout	determines the number of output arguments from a function

Functions	Input/Output (I/O) Control
disp	displays a string or a matrix in the command window
input	allow the user to enter values
load	loads matrices from a file
save	saves variables in a file

Functions	Comparison Operators
<	less than
<=	less than or equal to
>	greater than
>=	greater than or equal to
==	equal to
~=	not equal to

Special Characters Logical Operators	
&&	and
	or
~	not
xor	exclusive or

Functions	Control Structures
if	checks a condition resulting in either true or false
else	defines the path if the result of an if statement is false
elseif	defines the path if the result of an if statement is false, and specifies a new logical test
end	identifies the end of a control structure
switch	part of the case selection structure
case	sorts responses
otherwise	part of the case selection structure
for	generates a loop structure
break	causes the execution of a loop to be terminated
while	generates a loop structure
continue	terminates the current pass through a loop, but proceeds to the next pass
menu	creates a menu to use as an input vehicle

Functions	Logical Functions
find	determines which elements in a matrix meet the input criterion
all	checks to see if a criterion is met by all the elements in an array
any	checks to see if a criterion is met by any of the elements in an array

Functions	Special Matrices
ones	creates a matrix containing all ones
zeros	creates a matrix containing all zeros
eye	generates an identity matrix
magic	creates a "magic" matrix

Special	Characters Data Types
{ }	cell array constructor
' '	string data (character information)

Functions	Data Type Manipulation
single	changes an array to a single-precision array
double	changes an array to a double-precision array
uint8	8-bit unsigned integer
uint16	16-bit unsigned integer
uint32	32-bit unsigned integer
uint64	64-bit unsigned integer
int8	8-bit signed integer
int16	16-bit signed integer
int32	32-bit signed integer
int64	64-bit signed integer
char	creates a padded character array
num2str	converts a numeric array to a character array
str2num	converts a character array to a numeric array

Functions	Manipulation of Symbolic Expressions
sym	creates a symbolic variable, expression, or equation
syms	creates symbolic variables
expand	expands an expression or equation
factor	factors an expression or equation
collect	collects like terms
simplify	simplifies using Mupad's built-in simplification rules
simple	tries and reports all the simplification functions, and selects the shortest answer
solve	solves a symbolic expression or equation
diff	finds the symbolic derivative of a symbolic expression
int	finds the symbolic integral of a symbolic expression
ezplot	creates an $x - y$ plot of a symbolic expression