AARHUS UNIVERSITET
INGENIØRHØJSKOLEN

Kursus: M3NUM1

Eksamensdato: 04.06.2019 – 09:30 – 13:30 (forlænget prøvetid 14:30)

Eksamenstermin: Juni 2019

Underviser: Henning T. Søgaard

Praktiske informationer
Ingeniørhøjskolen udleverer:
4 stk. hvidt papir

Digital eksamen:
Opgaven tilgås og afleveres gennem den digitale eksamensportal.
Håndskrevne dele af opgavebesvarelsen skal digitaliseres og afleveres i den digitale eksamensportal. **Opgavebesvarelsen skal afleveres i PDF-format.**

Husk at uploade og aflevere i Digital eksamen til tiden. Du vil modtage en elektronisk afleveringskvittering, straks du har afleveret.

Husk at aflevere til tiden, da der ellers skal indsendes dispensationsansøgning

Husk angivelse af navn og studienr. på alle sider, samt i dokumenttitel / filnavn
Alle hjælpemidler må benyttes, herunder internettet som opslagsværktøj, men det er **IKKE** tilladt at kommunikere med andre digitalt.

Særlige bemærkninger: Det er kun muligt at aflevere elektronisk via Digital Eksamen portalen

OPGAVE 1

Antag, at integralet $\int_a^b f(t)\,dt$ ønskes udregnet. Resultatet af dette integral kan bestemmes ved at løse følgende differentialligning med den anførte begyndelsesbetingelse:

$$\frac{dy}{dt} = f(t),\ \ y(a) = 0$$

Løsningen til differentialligningen vil vise, hvordan y afhænger af t, og den opnåede værdi af y, når $t = b$, vil være lig med integralet $\int_a^b f(t)\,dt$.

I det følgende benyttes $f(t) = \cos(t^2)$, $a = 0$ og $b = 4$. Dvs. differentialligningen

$$\frac{dy}{dt} = \cos(t^2),\ \ y(0) = 0$$

skal løses fra $t = 0$ til 4.

(1a)

Færdiggør følgende MATLAB-script (erstat **<UDFYLD>** med korrekt MATLAB-kode), så det kan løse differentialligningen ved hjælp af Eulers metode fra $t = 0$ til 4 med skridt på $h = 0,4$.

```
clear
a = 0; b = 4; h = 0.4;
t = a:h:b;
y(1) = <UDFYLD>;
for i = 1:<UDFYLD>
  dydt = <UDFYLD>;
  y(i+1) = <UDFYLD>;
end
disp([t' y'])
```

Scriptet kan downloades fra Digital Eksamen under navnet `EulersMetode.m`.

Besvarelsen skal indeholde det færdiggjorte script samt det output i kommandovinduet, som scriptet producerer, når det køres. Angiv desuden ud fra resultatet det fundne estimat af integralet $\int_0^4 \cos(t^2)\,dt$.

(1b)

Anvend en fjerdeordens Runge-Kutta metode til at løse differentialligningen fra $t = 0$ til 4 med skridt på $h = 0,4$. Såvel talresultater som beregningsformler ønskes anført på tabelform.

i	t	y	k_1	t_{mid1}	y_{mid1}	k_2	t_{mid2}	y_{mid2}	k_3	t_{ende}	y_{ende}	k_4	ϕ
0													
1													
2													
3													
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Hvad er værdien af integralet $\int_0^4 \cos(t^2)\,dt$ ifølge løsningsresultatet?

(1c)

Følgende tabel viser den korrekte værdi af integralet $\int_0^4 \cos(t^2)\,dt$ samt estimater af integralet opnået vha. Eulers metode og en fjerdeordens Runge-Kutta metode, i begge tilfælde med $h = 0{,}2$. Tabellen viser desuden integralestimaternes afvigelser (fejl) i forhold til den korrekte værdi.

	Korrekt	Euler-estimat med $h = 0{,}2$	Runge-Kutta-estimat med $h = 0{,}2$
Integral	0,594460	0,798126	0,594405
Fejl	0,000000	0,203666	0,000055

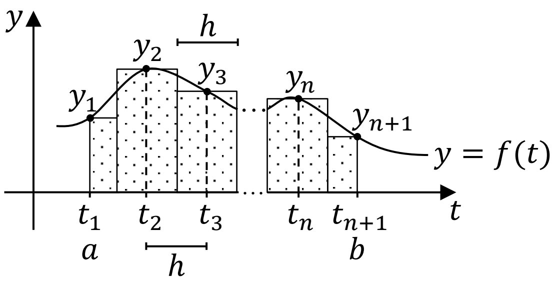
Eulers metode resulterer i størst fejl, men fejlen kan mindskes ved at mindske skridtlængden, h.

> Hvad skal h-værdien i Eulers metode ned på, for at man kan forvente, at fejlen kommer ned på $0{,}000055$, dvs. den fejlværdi, som Runge-Kuttas fjerdeordensmetode resulterer i ved $h = 0{,}2$?

(1d)

Der findes andre numeriske metoder til tilnærmet beregning af integralet $\int_a^b f(t)\,dt$. Fremgangsmåden i en af metoderne består i først at opdele intervallet fra a til b i n lige store delintervaller, som hver får bredden $h = (b-a)/n$ (fig. 1). Integralet, I, altså arealet under funktionens graf, kan derefter udregnes tilnærmet som summen af arealerne af en række rektangler:

$$I = \frac{h}{2}y_1 + hy_2 + hy_3 + \cdots + hy_n + \frac{h}{2}y_{n+1}$$
$$= \left(\frac{y_1}{2} + y_2 + y_3 + \cdots + y_n + \frac{y_{n+1}}{2}\right)h$$



Figur 1: Illustration af numerisk integration.

hvor $t_1, \dots, t_{n+1}$ er delepunkter mellem delintervallerne, og $y_1, \dots, y_{n+1}$ er funktionsværdierne svarende hertil ($y_i = f(t_i),\ i = 1, \dots, n+1$).

Her følger en næsten færdig MATLAB-funktion, `intgrl`, til numerisk beregning af integralet, `I`, ifølge ovenstående formel (kan downloades fra Digital Eksamen under navnet `intgrl.m`):

```
function I = intgrl(f,a,b,n)
h = (b-a)/n;
t = <UDFYLD>;
y = <UDFYLD>;
I = <UDFYLD>;
```

Inputargumenterne `f`, `a`, `b` og `n` svarer til henholdsvis $f(t)$, a, b og n som defineret tidligere.

> Færdiggør funktionen `intgrl` ved at erstatte hvert **<UDFYLD>** med korrekt MATLAB-kode. Ved beregning af `I` i sidste kodelinje kan den indbyggede MATLAB-funktion `sum`, der beregner summen af en vektors elementer, indgå i beregningsudtrykket. Afprøv den færdige funktion ved at beregne integralet $\int_0^4 \cos(t^2)\,dt$ med $n = 10$ delintervaller. Besvarelsen skal indeholde den færdiggjorte funktion samt anvendte MATLAB-kommandoer og opnåede resultater i forbindelse med afprøvningen.

OPGAVE 2

(2a)

Der skal skrives en MATLAB-funktion, `kaststat`, der givet et antal terningkast kan optælle, hvor mange henholdsvis 1'ere, 2'ere, 3'ere, 4're, 5'ere og 6'ere, der er blandt kastene. Hvis man fx efter 10 kast har slået henholdsvis 4, 4, 3, 5, 5, 6, 1, 2, 1 og 4 skal funktionen kunne nå frem til et optællingsresultat som illustreret med følgende MATLAB-kommando og tilhørende resultat i kommandovinduet:

```
>> stat = kaststat([4 4 3 5 5 6 1 2 1 4])
stat =
     2     1     1     3     2     1
```

Det ses, at de 10 kast skal kunne angives som vektorinput, og at output skal være en vektor, der fortæller, hvor mange 1'ere, 2'ere, …, 6'ere der er blandt de 10 kast – i dette tilfælde henholdsvis 2, 1, 1, 3, 2 og 1.

Her følger en ufærdig version af funktionen (kan downloades fra Digital Eksamen under navnet `kaststat.m`):

```
function stat = kaststat(kast)
<UDFYLD>
stat = zeros(1,6);
for <UDFYLD>
  <UDFYLD>
end
```

Funktionens kode skal fungere således, at en `for`-løkke trin for trin foretager optællingen, sådan at vektoren `stat` til slut indeholder optællingsresultaterne.

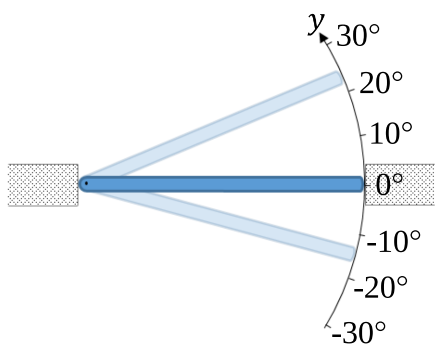
> Færdiggør funktionen ved at erstatte hvert **<UDFYLD>** med korrekt MATLAB-kode (nul, én eller flere linjer efter behov). Afprøv derefter funktionen med følgende tre vektorer som input (én efter én): `[5 2 5 3 1 2 5 3 1 3]`, `[2 2 5 6 3 5 5 1 4 3 6 1 3 3 3]` og `[]`. Besvarelsen skal indeholde den færdiggjorte funktion samt anvendte MATLAB-kommandoer og opnåede resultater i forbindelse med afprøvningen.

(2b)

Vi betragter en svingdør som illustreret i figur 2 (set ovenfra). Døren er fjeder- og dæmperbelastet, så den efter at have været åbnet en vis vinkel vil svinge tilbage til lukket stilling (0°) uden oversving. Antag, at døren til tidspunktet $t_1 = 0$ s begynder at svinge i fra en given vinkel, y_1. Vi betragter derefter dens vinkelstillinger, $y_2, y_3, y_4, \ldots$ (grader), til tidspunkterne $t_2 = 0{,}1$ s, $t_3 = 0{,}2$ s, $t_4 = 0{,}3$ s, … . Der gælder følgende model for dørens indsvingning til lukket stilling:

$$y_i = ay_{i-1} + by_{i-2}\,,\quad i = 3, 4, 5, \ldots$$

I denne opgave regnes med, at $a = 1{,}69$ og $b = -0{,}71$, samt at dørens to første vinkelstillinger er henholdsvis $y_1 = 83°$ og $y_2 = 82°$.



Figur 2: Svingdør.

Her følger et ufærdigt MATLAB-script, der på baggrund af indsvingningsmodellen skal fastlægge og plotte dørens vinkelstillinger, y_1, y_2, y_3, ..., som funktion af tiden, t_1, t_2, t_3, ... (defineret i form af vektorer `y` og `t`):

```
eps = 1;                % Grad
a = 1.69; b = -0.71;
dt = 0.1;               % Sekunder
y(1) = 83; y(2) = 82; % Grader
t(1) = 0;  t(2) = dt; % Sekunder
i = 2;
while <UDFYLD>
  i = <UDFYLD>;
  y(i) = <UDFYLD>;
  t(i) = <UDFYLD>;
  <UDFYLD>
end
plot(t,y)
xlabel('t (s)'), ylabel('y (gr.)')
```

Scriptet kan downloades fra Digital Eksamen under navnet `doerlukning.m`.

> Færdiggør scriptet ved at erstatte hvert **<UDFYLD>** med korrekt MATLAB-kode, idet det oplyses, at `while`-løkken skal stoppe, så snart der nås en vinkelstilling, y_i, der afviger mindre end $\varepsilon = 1°$ fra lukket stilling (0°). Koden skal være lavet, så den også fungerer korrekt, hvis startvinklerne er negative, fx $y_1 = -83°$ og $y_2 = -82°$. Besvarelsen skal indeholde det færdiggjorte script samt det plot, der produceres, når det køres.

(2c)
Vi betragter atter svingdøren fra opgave (2b) (se figur 2). Lad igen y betegne dørens vinkel, nu i radianer, og t betegne tiden i sekunder. Idet det antages, at døren til tidspunktet $t = 0$ s begynder at svinge i fra en given vinkel, y_0, og at vinkelhastigheden til dette tidspunkt er nul, gælder der følgende differentialligning og begyndelsesbetingelser for $y(t)$:

$$Iy''(t) + cy'(t) + ky(t) = 0, \quad y(0) = y_0, \quad y'(t) = 0$$

I denne opgave regnes med:

$$I = 7{,}2 \text{ kg m}^2, \; c = 25 \text{ N m/(rad/s)}, \; k = 18 \text{ N m/rad} \;\text{ og }\; y_0 = 1{,}45 \text{ rad}$$

I denne opgave skal man anvende MATLAB-funktionen `rk4system`, som har været anvendt ved undervisningen til løsning af differentialligningssystemer vha. en fjerdeordens Runge-Kutta metode. Funktionen kan downloades fra Digital Eksamen under navnet `rk4system.m`.

> Omform differentialligningen til et system af to koblede førsteordens differentialligninger ved at indføre vinkelhastigheden $\omega(t) = y'(t)$ som hjælpevariabel. Benyt derefter `rk4system` til at løse differentialligningen i tidsrummet $t = 0$ til 5 s med tidsskridt på $h = 0{,}5$ s. Plot de fundne løsningsværdier (y som funktion af t). Besvarelsen skal indeholde omformningen til førsteordens differentialligninger, de anvendte MATLAB-kommandoer, output fra `rk4system` og det nævnte plot.

OPGAVE 3

Figur 3 viser et rektangulært stålemne med to kvadratiske huller. Emnet er fast indspændt i begge ender og er i figuren opdelt i fem sektioner (1, 2, …, 5), som er afgrænset af seks knuder, **1**, **2**, …, **6**. Hver sektion kan betragtes som et stangelement. Sektionerne har følgende længder og tværsnitsarealer:

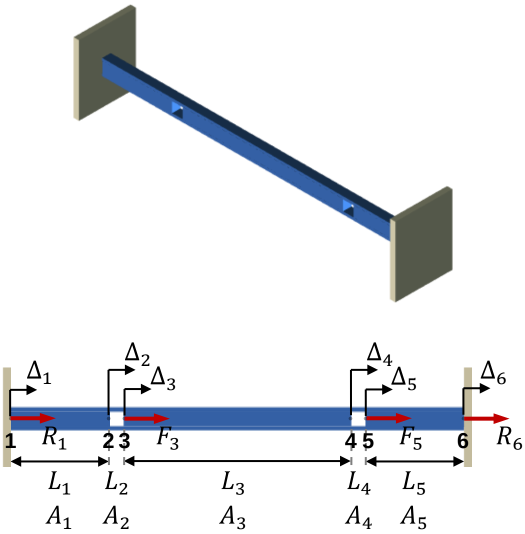
$$L_1 = L_5 = 130 \text{ mm}$$

$$L_2 = L_4 = 20 \text{ mm}$$

$$L_3 = 300 \text{ mm}$$

$$A_1 = A_3 = A_5 = 450 \text{ mm}^2$$

$$A_2 = A_4 = 150 \text{ mm}^2$$



Figur 3: Fast indspændt stålemne i 3D (øverst) og som plantegning (nederst).

Emnet, hvis E-modul er $E = 210$ GPa, er i knuderne **3** og **5** påvirket af eksterne kræfter på henholdsvis $F_3 =$ 19,4 kN og $F_5 = 54{,}3$ kN som vist i figuren. Knude **1** og **6** er fikserede og dermed påvirket af reaktionskræfter, som er benævnt henholdsvis R_1 og R_6. Kraftpåvirkningerne vil bevirke, at knude **2** til **5** udsættes for forskydninger, $\Delta_2, \ldots, \Delta_5$, mens knuderne **1** og **6**, som er fikserede, ikke forskydes ($\Delta_1 = \Delta_6 = 0$). Der regnes positiv retning mod højre.

PS! Det er *ikke* et krav, at der anvendes MATLAB til løsning af denne opgave.

(3a)

Beregn stivhedsmatricen, K, for det sektionsopdelte stålemne. Anfør resultatet med enhed.

(3b)

Beregn forskydningerne Δ_2, Δ_3, Δ_4 og Δ_5, og anfør resultaterne med enheder.

PS! Hvis spørgsmål (3a) ikke er besvaret, kan man antage, at følgende stivhedsmatrix er gældende:

$$K = \begin{bmatrix} 0{,}9 & -0{,}9 & 0 & 0 & 0 & 0 \\ -0{,}9 & 1{,}9 & -1{,}0 & 0 & 0 & 0 \\ 0 & -1{,}0 & 1{,}3 & -0{,}3 & 0 & 0 \\ 0 & 0 & -0{,}3 & 1{,}3 & -1{,}0 & 0 \\ 0 & 0 & 0 & -1{,}0 & 1{,}9 & -0{,}9 \\ 0 & 0 & 0 & 0 & -0{,}9 & 0{,}9 \end{bmatrix} \text{GN/m}$$

(3c)

Hvor meget forlænges sektionen mellem knude **2** og **3** (sektion 2) ifølge resultatet af spørgsmål (3b)? Angiv resultatet i såvel mm som procent.

PS! Hvis spørgsmål (3b) ikke er besvaret, kan man antage, at følgende forskydninger er gældende:

$$\Delta_2 = 0{,}024 \text{ mm}, \;\; \Delta_3 = 0{,}046 \text{ mm}, \;\; \Delta_4 = 0{,}055 \text{ mm}, \;\; \Delta_5 = 0{,}057 \text{ mm}$$

OPGAVE 4

Der er givet følgende ligningssystem med ubekendte x_1, x_2 og x_3:

$$
\begin{aligned}
2(x_1 + 3x_2)^2 x_3^2 &= 18 \\
4x_1x_2 + 2x_1 - 4x_3^3 &= -14 \\
x_1 + 3x_2 - 2x_3 &= -1
\end{aligned}
$$

(4a)

Der defineres en søjlevektor **X** bestående af de ubekendte i ligningssystemet: $\mathbf{X} = (x_1 \;\; x_2 \;\; x_3)^{\mathrm{T}}$.

> Er det muligt at opskrive ligningssystemet på matrixform, $\mathbf{AX} = \mathbf{B}$, hvor **A** og **B** er henholdsvis en 3×3 matrix og en 3×1 matrix, begge indeholdende konstanter? Begrund svaret. Hvis svaret menes at være ja, opskriv da **A** og **B**.

(4b)

En løsning til ligningssystemet ønskes fundet ved hjælp af Newton-Raphsons metode, idet der anvendes startværdier på $x_1 = 2,\ x_2 = 1$ og $x_3 = 1$. Her følger et ufærdigt MATLAB-script til formålet:

```
X = [2; 1; 1]
for i = 1:<UDFYLD>
  f = [2*(X(1)+3*X(2))^2*X(3)^2 - 18
       <UDFYLD>
       X(1) + 3*X(2) - 2*X(3) + 1];
  J = [4*(X(1)+3*X(2))*X(3)^2, 12*(X(1)+3*X(2))*X(3)^2, <UDFYLD>
       4*X(2)+2,               <UDFYLD>,               -12*X(3)^2
       <UDFYLD>,               3,                      -2];
  X = X - <UDFYLD>
end
```

Scriptet kan downloades fra Digital Eksamen under navnet `NewtRaphs.m`.

> Færdiggør scriptet ved at erstatte hvert **<UDFYLD>** med korrekt MATLAB-kode, idet det oplyses, at løsningen med hensyn til $x_1,\ x_2$ og x_3 ønskes bestemt med fire sikre decimaler. Besvarelsen skal indeholde det færdiggjorte script, den fundne løsning med fire sikre decimaler samt en begrundelse for, at man ud fra scriptets resultater kan være sikker på de fire decimaler.

(4c)

Der betragtes nu et andet ligningssystem med ubekendte u, v og w:

$$
\begin{aligned}
-5u + 2v - w &= -19 \\
6u - 4v - 3w &= -3 \\
-u + 3v + 2w &= 15
\end{aligned}
$$

Dette ligningssystem har én løsning, nemlig $[u, v, w] = [4, 3, 5]$.

> Hvis man anvender Newton-Raphsons metode til løsning af ligningssystemet, opnår man denne løsning allerede efter første iteration, uanset hvilke startværdier man anvender. Forklar, hvorfor der kun skal gennemføres én iteration for at finde løsningen?