Tentamen för Matematisk överbryggningskurs

Tid och plats: 2018-08-29, kl 8:30-12:30, L. **Ansvarig:** Jacques Huitfeldt, 031-7721093.
Tentamensrond: Carl Lundholm, tel. 031-7726792.

Betygsgränser: 20, 30 resp. 40 poäng. Tentan omfattar totalt 50 poäng.

Hjälpmedel: Inga hjälpmedel tillåtna, utom bifogat formelblad.

Uppgift 1. En viss funktion $s(t)$ är positiv (dvs. $s(t) > 0$ för alla t) och ges av sambandet

$$s(t) + \ln(s(t)) = 1 - t$$

Vi skall rita funktionens graf över intervallet $0 \leq t \leq 10$.

(a). Skriv den kod i Matlab som behövs för att med hjälp av `fzero` rita upp grafen av $s(t)$.

(b). Derivera sambandet för funktionen $s(t)$ så att vi får en differentialekvation för $s(t)$. Skriv den kod i Matlab som behövs för att med hjälp av `ode45` lösa differentialekvationen och rita upp grafen av $s(t)$.

Hela uppgiften ger maximalt **(8p)**

Uppgift 2. Visa att om $\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_r$ är egenvektorer till en matris $\boldsymbol{A}$ och sammanhörande egenvärden $\lambda_1, \lambda_2, \cdots, \lambda_r$ alla är olika så är $\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_r$ linjärt oberoende.

(8p)

Uppgift 3. Betrakta följande linjära system av ODE

$$\begin{cases} \boldsymbol{u}' = \boldsymbol{A}\boldsymbol{u}, \; t > 0 \\ \boldsymbol{u}(0) = \boldsymbol{u}_0 \end{cases}, \quad \boldsymbol{A} = \begin{bmatrix} -4 & 1 \\ 3 & -2 \end{bmatrix}, \quad \boldsymbol{u}_0 = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$$

(a). Lös problemet med egenvärdesmetoden.

(b). Härled formeln du använde i **(a)** utgående från att $\boldsymbol{A}$ är diagonaliserbar ($\boldsymbol{V}^{-1}\boldsymbol{A}\boldsymbol{V} = \boldsymbol{D}$).

(c). Tag ett steg med Euler framåt med steglängden $h = 0.1$.

Hela uppgiften ger maximalt **(8p)**

Uppgift 4. Betrakta det icke-linjära ekvationssystemet

$$\begin{cases} x_1 + x_2 \exp(x_1 x_2) = -1 \\ x_1^3 + x_2 = 1 \end{cases}$$

(a). Skriv systemet på kompakt form $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$. Vad blir $\boldsymbol{f}(\boldsymbol{x})$ och vad blir $\boldsymbol{Df}(\boldsymbol{x})$, dvs. beräkna derivatamatrisen.

(b). Härled Newtons metod för ekvationen $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{0}$.

(c). Tag ett steg med Newtons metod utgående från startapproximationen $\boldsymbol{x}_0 = (0, 1)$ för hand, dvs. utför beräkningarna med papper och penna.

(d). Skriv ned den kod i Matlab som utgående från startapproximationen $\boldsymbol{x}_0 = (0, 1)$ beräknar lösningen med Newtons metod.

Hela uppgiften ger maximalt **(8p)**

Uppgift 5. Vi skall bestämma stationära punkter till funktionen

$$f(\boldsymbol{x}) = -x_1^3 + 5x_1x_2 - 3x_2^3 - 1$$

i området $-4 \leq x_1 \leq 4$, $-4 \leq x_2 \leq 4$.

(a). Beräkna gradienten $\boldsymbol{\nabla} f(\boldsymbol{x})$ och Hessematrisen $\boldsymbol{H}(\boldsymbol{x})$ för hand. Använd matris- och vektorbeteckningar.

(b). Skriv ned den kod i MATLAB som behövs för att lokalisera de stationära punkterna. Rita dels funktionsyta med `surf`, dels nivåkurvor med `contour`. Antag att vi genom att inspektera funktionsytan ser att de intressanta nivåerna ligger mellan -40 och 40. Använd detta vid uppritning av nivåkurvorna. Rita även i samma figur noll-nivåkurvorna till komponenterna i gradienten.

(c). Skriv ned den kod i MATLAB som, utgående från en startapproximation, beräknar stationär punkt med `fsolve` och sedan med `eig` avgör dess typ.

Hela uppgiften ger maximalt **(8p)**

Uppgift 6. Temperaturen i en oisolerad glödande tråd beskrivs av

$$\begin{cases} u'_t - \kappa u''_{xx} = f(x,t) + C(u_{omg}^4 - u^4), \; 0 < x < L, \; 0 < t \leq T \\ u(0,t) = g_0, \; u(L,t) = g_L, \; t > 0 \\ u(x,0) = u_0(x), \; 0 < x < L \end{cases}$$

(a). Vi skall lösa problemet med linjemetoden. Skriv ned det begynnelsevärdesproblem vi får då vi approximerar x-derivatorna med differenskvoter. Vi vill ha svaret på vektorform.

(b). Skriv ned den kod som i MATLAB behövs för att med `ode45` beräkna och sedan rita upp temperaturen under T sekunder. Antag att konstanterna $\kappa, C, u_{omg}, L, g_0, g_L$ redan givits värden (variablerna `kappa`, `C`, `uomg`, `L`, `g0`, `gL`) samt att funktionerna $f(x,t), u_0(x)$ redan är definierade (enligt `f=@(x,t)...`, `u0=@(x)...`).

Hela uppgiften ger maximalt **(10p)**

Differensapproximationer

$$D_+u(x) = \frac{u(x+h)-u(x)}{h} = u'(x) + \mathcal{O}(h), \qquad D_-u(x) = \frac{u(x)-u(x-h)}{h} = u'(x) + \mathcal{O}(h)$$

$$D_0u(x) = \frac{u(x+h)-u(x-h)}{2h} = u'(x) + \mathcal{O}(h^2)$$

$$D_+D_-u(x) = \frac{u(x+h)-2u(x)+u(x-h)}{h^2} = u''(x) + \mathcal{O}(h^2)$$

$$\frac{u(x+2h)-2u(x+h)+2u(x-h)-u(x-2h)}{2h^3} = u'''(x) + \mathcal{O}(h^2)$$

$$\frac{u(x+2h)-4u(x+h)+6u(x)-4u(x-h)+u(x-2h)}{h^4} = u^{(4)}(x) + \mathcal{O}(h^2)$$

Taylorutveckling

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots$$

$$+ \frac{f^{(n)}(a)}{n!}(x-a)^n + R_{n+1}(x)$$

$$R_{n+1}(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-a)^{n+1} = \mathcal{O}((x-a)^{n+1})$$

Taylorutveckling i två variabler

$$f(x,y) = f(a,b) + f'_x(a,b)h + f'_y(a,b)k + \tfrac{1}{2}\big(f''_{xx}(a,b)h^2 + 2f''_{xy}(a,b)hk + f''_{yy}(a,b)k^2\big) + \cdots$$

där $h = x - a$ och $k = y - b$.

Lite stolpar i MATLAB som stöd för minnet.

Elementära funktioner

```
abs(x), sqrt(x), exp(x), log(x), cos(x), sin(x), tan(x)
```

Matris- och vektorfunktioner

```
size(A), length(v)
sum(v), prod(v), max(v), min(v), sort(v), norm(v), dot(u,v)
```

Matrisuppbyggande funktioner

```
A=ones(m,n), A=zeros(m,n), A=eye(m,n), A=diag(d,k), A=spdiags(D,K,m,n)
```

Villkorssatser

```
if uttryck          if uttryck              if uttryck
    satser              satser                  satser
end                 else                    elseif uttryck
                        satser                  satser
                    end                     end
```

Repetitionssatser

```
while uttryck      for variabel=uttryck      for variabel=start:steg:slut
    satser             satser                    satser
end                end                       end
```

Egna funktioner

```
function ut=funktionsnamn(parametrar)       handtagsnamn=@(parametrar) sats
    satser
```

Beräkningsfunktioner

```
x=fzero(f,x0), x=fminbnd(f,x0,x1), q=integral(f,a,b), [t,U]=ode45(f,tspan,u0)
x=A\b, [V,D]=eig(A), [V,D]=eigs(A,K,'SM')
x=fsolve(f,x0), x=fminunc(f,x0), q=integral2(f,a,b,c,d)
```

Grafik

```
x=linspace(a,b,n)
plot(x,y), plot3(x,y,z), quiver(u,v,du,dv,sc), quiver3(...)
[X,Y]=meshgrid(x,y)
surf(X,Y,Z), contour(X,Y,Z,lv)
```