Lösningsanvisningar till vissa av de icke obligatoriska workout-uppgifterna i Beräkningsvetenskap II

Kurvanpassning

6.
```
A = [1 1;
     2 1;
     1 2;
     2 3;
     2 5;
     2 4];

v = [30.006;
     44.013;
     46.006;
     76.012;
     108.010;
     92.001];

ATA = A'*A
ATv = A'*v

atomvikter = ATA\ATv
```

7. Interpolationspolynomet genom (t_k, y_k), (t_{k+1}, y_{k+1}): $p_1(t) = a_0 + a_1(t - t_k)$, där $a_0 = y_k$ och $a_1 = (y_{k+1} - y_k)/(t_{k+1} - t_k)$. Lägg till punkten (t_l, y_l). Interpolationspolynomet genom (t_k, y_k), (t_{k+1}, y_{k+1}), (t_l, y_l): $p_2(t) = p_1(t) + a_2(t - t_k)(t - t_{k+1})$, där

$$a_2 = \frac{y_l - y_k}{(t_l - t_k)(t_l - t_{k+1})} - \frac{y_{k+1} - y_k}{(t_{k+1} - t_k)(t_l - tk + 1)}$$

Enligt uppgiften skull felet för ett givet t-värde $t = c$ uppskattas med $p_2(c) - p_1(c) = a_2(c - t_k)(c - t_{k+1})$. Denna formel tillämpad på det speciella fall som nämns i uppgiften: med $t_k = 250$, $t_{k+1} = 300$, $t_l = 200$ blir det uppskattade felet för $c = 260$ ca -4.3.

8. Gör ansatserna: $f(x) = a_0 + a_1 x$, $g(x) = f(x) + c = c + a_0 + a_1 x$. Ställ nu upp ett överbestämt ekvationssystem med $N + M$ ekvationer för de 3 obekanta c, a_0 och a_1. Ekvationerna i systemet utgörs av villkoren $f(x_i) = \tilde{f}(x_i), i = 1, \ldots, N$, samt $g(x_i) = \tilde{g}(x_i), i = 1, \ldots, M$. Bilda sedan normalekvationerna och lös dem för att få fram värden på de tre obekanta koefficienterna.

9. Inför beteckningen $x = 1/r^2$. Den hypotes som ska testas kan nu uttryckas som ansatsen $p = cx$, där det gäller att bestämma c. Samla

in data (x samt p) för ett antal bostadsrätter som ungefär samtidigt sålts i Uppsala. Bestäm sedan c för dessa data genom minstakvadrat-approximation.

10. Se läroboken, där detta gås igenom detaljerat.

ODE, del 1

5. `function y = euler(f,a,b,y0,n);`

```
    h = (b-a)/n;
    t = a:h:b;

    y = [];
    ynow = y0; % Begynnelsevärde
    for i = 1:n-1
        ynext = ynow + h*feval(f,t(i),ynow);
        y = [y; ynext];
        ynow = ynext;
    end
```

6. Utparameterna `y` blir en matris. Låt rad k i `y` innehålla lösningen för tidssteg k. Om funktionen `f` returnerar en radvektor, så kommer `ynext` att bli en radvektor och i så fall krävs *inga* ändringar i funktionen `euler`. Om funktionen `f` istället returnerar en kolonnvektor (vilket Matlabs *inbyggda* ODE-lösare förutsätter), så måste raden `y = [y; ynext]` ändras till `y = [y; ynext']` (det vill säga `ynext` transponeras till en radvektor och läggs sedan in som ny rad i `y`.

7. Lösningsskiss: Först skriver man om modellen för pendeln som ett system av två ODE av första ordningen. Till detta system lägger man den givna ekvationen för $r'(t)$. Det resulterande systemet med tre första ordningens ODE löses sedan med Heuns metod (se läroboken).

ODE, del 2

OBS! Uppgifterna felnumrerade i uppgiftsbladet. Uppgift nr 7 borde vara nr 5, etc. Nedan används den korrigerade numreringen.

5. Tillämpa Heuns metod på testekvationen, $y' = \lambda y$, ger stabilitetsvillkoret $|1 + \lambda h + (\lambda h)^2/2| < 1$.

6. Enligt den ledning som ges i uppgiftstexten ska man uppskatta felet för steglängd h respektive $h/2$ genom att betrakta lösningen för $h/8$ som "rätt" svar. Låt $\epsilon(h)$ beteckna felet för steglängd h. Om man antar att $\epsilon(h) \approx ch^q$, där c är en konstant, så följer att kvoten $\epsilon(h)/\epsilon(h/2) \approx 2^q$. Genom att räkna ut denna kvot för de olika raderna i tabellen kan man alltså få en uppskattning av noggrannhetsordningen q.

7. (a) Explicit om $c_2 = 0$. Implicit om $c_2 \neq 0$.

   (b) Det gäller att undersöka lokala trunkeringsfelet. Notera att enligt differentialekvationen är $g(t_k, P_k) = P'(t_k)$ och $g(t_{k+1}, P_{k+1}) = P'(t_{k+1})$. Sätt in detta i uttrycket för "familjen" av metoder. Taylorutveckla sedan kring t_k. Sortera sedan termerna i uttrycket efter h-potens. För noggrannhetsordning större än eller lika med 1 krävs att alla förstagradstermer i h försvinner. Detta uppnås om $c_1 + c_2 = 0$. Om man sätter $c_1 = c_2 = 0.5$, så försvinner även andragradstermerna i h. Detta ger noggrannhetsordning 2 vilket är största möjliga noggrannhetsordning i detta fall.

8. Enligt antagandena i uppgiften kan vi bortse från stabilitetsvillkoren. Det kommer att vara noggrannhetskraven som avgör hur litet h vi behöver välja. Med Euler framåt krävs steglängd $h = ch_0$, där $0 < c < 1$, medan Heuns metod kan använda $h = h_0$. Vi ska finna det värde på c för vilket metoderna ger samma exekveringstid. Exekveringstiden t kan uttryckas som $t = ns$, där n är antalet tidssteg och s är exekveringstiden per tidssteg. Vidare är $s = w\tau$, där w är antalet "operationer" per tidssteg och τ är exekveringstiden per operation. För Euler framåt och Heuns metod kommer den tidskrävande operationen att vara beräkning av differentialekvationens högerled $f(t, y)$. Jämfört med beräkningen av högerledet kommer övriga räkneoperationer i ett tidssteg att ta försumbar tid.

   Nu jämför vi metoderna. Värdet på τ är oberoende av metod, så det avgörande är hur metoderna skiljer sig åt beträffande n och w. Värdet på $n = T/h$ (om vi antar att starttiden är $t = 0$ och sluttiden är $t = T$). För Euler framåt blir detta $n_e = T/(ch_0)$. För Heun får vi $n_h = T/h_0 = cn_e$.

   Värdet på w är antalet funktionsanrop per tidssteg. För Euler framåt: $w_e = 1$. För Heun: $w_h = 2$.

Vi kan nu jämföra tiden för Euler respektive Heun: $t_e = n_e w_e \tau = n_e \tau$, $t_h = n_h w_h \tau = c n_e 2\tau = 2c t_e$. Slutsatsen blir att för lika exekveringstid krävs att $c = 0.5$. Om $c > 0.5$ lönar det sig att använda Euler framåt. Om $c < 0.5$ kommer det att gå snabbare att lösa problemet med Heuns metod.