

TANA17 Matematiska beräkningar med Matlab

Datorlektion 6. Text och filer

1 Textsträngar

Uppgift 1.1 Skapa en sträng som innehåller texten: `kommer du snart?`. Använd `length` för att kontrollera hur många tecken strängen innehåller. Ändra sedan ordet `du` till `ni` genom att ändra dessa tecken i strängen. Använd funktionen `disp` för att skriva ut er sträng på skärmen och kontrollera att ni gjort rätt. ☐

Uppgift 1.2 Man kan använda kommandot `input` för att läsa in enkla strängar. Skriv ett program som frågar efter ett namn och sedan skriver ut ett meddelande enligt `Jag tror också att <namn> vinner!`. ☐

Uppgift 1.3 Skriv en funktion `ReverseString` med en sträng som inparameter och samma sträng fast med tecknen i omvänd ordning som utparameter. Du skall alltså kunna skriva

```
>> ReverseString( 'Hej på dig!' )
ans =
    !gid åp jeH
```

Använd en repetitionssats som kopierar ett tecken i taget till en ny sträng. ☐

Uppgift 1.4 Skriv ett program som frågar efter ett reellt tal med `input` och sedan beräknar exponentialfunktionen. Följande utskrift skall ske:

```
Ge ett reellt tal? -4.7
exp(-4.70)=9.095e-03
```

Försök få utskriften exakt lika som exemplet. ☐

Uppgift 1.5 Skapa två strängar `str1='abcdef'` och `str2='AbCd12'`. Vilket resultat fås om du skriver `str1==str2`? Vad blir resultatet om du istället skriver `strcmp(str1,str2)`? ☐

Uppgift 1.6 Skriv en funktion `CountLetters` med en sträng som inparameter som räknar hur många bokstäver en given sträng innehåller. Använd Matlab funktionen `isletter`. Funktionen skall ge utskriften `"Strängen innehåller <antal> bokstaver"`. ☐

Uppgift 1.7 En vektor kan skrivas ut till skärmen med kommandot `disp`. Det blir inte särskilt snyggt om elementen i vektorn har väldigt olika storleksordning. Vi vill därför förbättra utskriften genom att skriva en funktion `displayVector` som skriver ut vektorn på följande sätt

```
>> displayVector( 'x' , [-2.845 0.323 12.3*10^-6 1002.4] )
x=( -2.85e+00 , 3.23e-01 , 1.23e-05 , 1.00e+03 )
```

Vi vill alltså skriva varje element på exponentform med två decimaler. $\square$

Uppgift 1.8 (Svår) En rot till ekvationen $f(x) = xe^x - 1 = 0$ kan hittas genom att beräkna gränsvärdet för talföljden,

$$\begin{cases} x_0 = 1, \\ x_n = e^{-x_{n-1}}, \quad n = 1, 2, 3, \dots \end{cases}$$

Skriv ett program som beräknar talföljdens gränsvärde. Du skall fortsätta att bilda nya tal x_n ända tills avbrottskriteriet $|x_n - x_{n-1}| < 10^{-4}$ är uppfyllt.

Vi vill även studera konvergenshastigheten. Ditt program skall därför skriva en tabell innehållande iterationsnummer n , x_n , samt skillnaden $x_n - x_{n-1}$ till skärmen. Utskriften skall göras enligt följande mönster:

n	x(n)	x(n)-x(n-1)

1	0.3678794	-6.32e-01
2	0.6922006	3.24e-01
3	0.5004735	-1.92e-01

2 Filhantering

I Matlab finns kommandon för att spara det man arbetar med så att man kan fortsätta senare. De enklaste kommandona är `save` och `load`. Dessa kan används för att spara/återskapa data man arbetar med. Skriver man

```
>> save sparade_data.mat
```

sparas alla variabler man för närvarande har i minnet till filen `sparade_data.mat`. Senare kan man skriva

```
>> load sparade_data.mat
```

för att få tillbaka det man arbetat med. För att se vilka variabler man för närvarande har i arbetsminnet finns två kommandon `who` och `whos`. Det finns även ett kommando `clear` som rensar arbetsminnet på innehåll. Prova att spara dina variabler i en fil, rensa arbetsminnet, och återskapa dina data!

Kommandona `load` och `save` är främst användbara för att kunna spara pågående arbete för att fortsätta senare.

Uppgift 2.1 Skriv `clear` för att rensa Matlabs minne. Skriv sedan `who` eller `whos` för att kontrollera att det verkligen är tomt. Skapa variablerna `x=1` och `y=2`. Spara sedan endast `x` genom att skriva

```
>> save x_variabeln.mat x
```

Rensa sedan minnet med `clear` och ladda in `x` igen genom att skriva

```
>> load x_variabeln.mat
```

Kontrollera med `who` att det fungerade. $\square$

Uppgift 2.2 En spelbricka kan beskrivas med en 5×5 matris vars element är slumpmässiga heltal mellan 0 och 99. Skriv ett program som skapar en slumpmässig bricka och skriver ut den till en textfil. Utskriften skall ske enligt följande mönster:

```
*****
* 12 * 0 * 25 * 89 * 76 *
*****
* 17 * 63 * 7 * 9 * 1 *
*****
* 85 * 18 * 82 * 3 * 11 *
*****
* 12 * 28 * 43 * 99 * 73 *
*****
* 93 * 16 * 57 * 91 * 38 *
*****
```

Försök att få utseendet så likt exemplet som möjligt. $\square$

Uppgift 2.3 Skriv ett program som skriver en tabell över exponentialfunktionens värden till en fil. Utskriften skall ske enligt följande mönster:

```
*****
*      x      * exp(x) *
*****
* 0.00 * 1.000000 *
* 0.10 * 1.105171 *
* 0.20 * 1.221403 *
*      :      *
*      :      *
* 0.90 * 2.459603 *
* 1.00 * 2.718282 *
*****
```

Försök att få din utskrift så lik tabellen ovan som möjligt. $\square$

Uppgift 2.4 Låt N och M vara två heltal. Skapa en $N \times M$ matris A med slumpstal. Skriv sedan ut matrisens element i en fil. Elementen skall skrivas ut radvis. $\square$

Uppgift 2.5 (Svår) Kvadratroten av ett positivt tal a kan beräknas genom att gränsvärdet till talföljden

$$\begin{cases} x_1 = 1, \\ x_{n+1} = \frac{1}{2} \left(x_n + \frac{a}{x_n} \right), \quad n = 1, 2, \dots \end{cases}$$

bestäms. Tillräckligt noggrannhet anses ha uppnåtts då $|x_{n+1} - x_n| < 10^{-9}$.

- Skriv en funktion `kvadratrot` med ett tal a som inparameter och en approximation av $\sqrt{a}$ som utparameter. Använd din funktion för att beräkna $\sqrt{3}$.
- Använd din funktion ifrån uppgift (a) för att skapa en tabell över $\sqrt{x}$, för $x = 1, 2, 3, \dots, 100$. Din tabell skall skrivas till en textfil. Du skall börja med att skriva ett tabellhuvud enligt följande mönster

```

      x              sqrt(x)
-----
```

en typisk rad i tabellen skall sedan se ut på följande sätt

```

59              7.68114575
```

Om du inte har löst (a) får du använda Matlabs inbyggda funktion `sqrt` då du löser (b). $\square$

TANA17 Matematiska beräkningar med Matlab

Facit till Datorlektion 6.

1 Textsträngar

Uppgift 1.1 Använd följande MATLAB kommandon:

```
>>str = 'kommer du snart?';  
>>str(8:9)='ni';  
>>disp(str)
```

`length(str)` ger svaret 16.

Uppgift 1.2 Din scriptfil kan exempelvis innehålla kommandona:

```
namn=input('Vem tror du vinner?', 's');  
str=['Jag tror också att ', namn , ' vinner!'];  
disp(str);
```

Uppgift 1.3 På filen `ReverseString.m` skriver vi

```
function [OutString]=ReverseRtring( InString )  
  
    n=length(InString);  
    OutString=blanks(n); % Tom sträng  
    for i=1:n  
        OutString(i)=InString(n-i+1);  
    end  
end
```

Vi kan sedan använda funktionen genom att skriva

```
>> ReverseString( 'Hej' )  
ans =  
    jeH
```

Uppgift 1.4 Vi skriver på filen `TestUppgift.m`

```
x=input('Ge ett reellt tal?');  
fprintf(1,'exp(%5.2f)=7.3e\n',x,exp(x) )
```

Uppgift 1.5 Eftersom bara andra och fjärde tecknet i de bägge strängarna är lika kommer resultatet då vi skriver `str1 == str2` att bli en radvektor (0 1 0 1 0 0). Kommandot `strcmp(str1,str2)` ger resultatet 0 (*falskt*) eftersom de båga strängarna är olika.

Uppgift 1.6 På filen `CountLetters.m` skriver vi

```
function [Antal]=CountLetters( str )

n=length(str);
Antal=0;
for i=1:n
    if isletter( str(i) )
        Antal=Antal+1;
    end
end
fprintf(1,'Strängen innehåller %i bokstäver\n',Antal)
end
```

Vi kan sedan använda funktionen genom att skriva

```
>> CountLetters('abc 123 de')
Strängen innehåller 5 bokstäver
ans =
     5
```

Uppgift 1.7 På filen `displayVector.m` skriver vi

```
function displayVector( name , vector )

n=length(vector);

% skriv först ut början: <name>=(
fprintf(1,' %s=(',name);

% skriv nu ut ett elemnt i taget med , och mellanslag.
for i=1:n-1
    fprintf(1,' %7.2e ',vector(i));
end

% skriv nu ut sista elementet utan , efter.
fprintf(1,' %7.2e )\n',vector(n) );
end
```

Uppgift 1.8 På filen hittaRot.m skriver vi

```
% Inför variabler x0 och x1 som representerar två på varandra
% följande approximationer
x0=-1;x1=1;n=0;    % Med x0=-1 blir avrottsvikkloret falskt
tol=10^-4;
fprintf(1,'  n      x(n)      x(n)-x(n-1)\n');
fprintf(1,'-----\n');
while abs(x0-x1)>tol
    % beräkna nästa approximation
    x0=x1;
    x1=exp(-x0);
    n=n+1;

    % Skriv ut en rad i tabellen
    fprintf(1,' %i    %9.7f  %9.2e\n',n,x1,x1-x0);
end
```

2 Filhantering

Uppgift 2.1 Skriv i terminalen

```
>> clear
>> whos
>> x=1;y=2;
>> save x_variabeln.mat x
>> clear
>> whos          % Nu är minnet tomt!
>> load x_variabeln
>> who           % Nu finns x igen.
Your variables are:
    x
```

Uppgift 2.2 På filen SpelBricka.m skriver vi

```
function SpelBricka( filnamn )

N=5;
fid=fopen( filnamn , 'w' );
for i=1:N
    % Skriv ut en rad. Först **** och sedan siffrorna.
    fprintf(fid,'*****\n')
    for j=1:N
        % skriv ut en *, mellanslag och sedan siffran
        S=randi([0 99],1,1);
```

```

        fprintf(fid,'* %2i ',S);
    end
    fprintf(fid,'*\n') % sista * på raden

end
fprintf(fid,'*****\n')
fclose(fid)
end

```

Anropar vi med `SpelBricka('test.txt')` kan vi titta på filen genom att skriva `type test.txt`.

Uppgift 2.3 Följande program skapar tabellen:

```

x=(0:0.1:1);
fid=fopen('tabell.txt','w');
fprintf(fid,'*****\n');
fprintf(fid,'*      x      * exp(x) * \n');
fprintf(fid,'*****\n');
for k=1:length(x)
    fprintf(fid,'*   %4.2f   * %8.6f * \n',x(k),exp(x(k)));
end;
fprintf(fid,'*****\n');
fclose(fid);

```

Uppgift 2.4 Använd exempelvis följande MATLAB rader:

```

N=20;M=7;
A=rand(N,M);
fid=fopen('matris.txt','w');
for i=1:N
    for j=1:M
        fprintf(fid,'%f ',A(i,j));
    end
    fprintf(fid,'\n');
end
fclose(fid);

```


Uppgift 2.5 På filen kvadratrot.m skriver vi

```
function [x1]=kvadratrot( a )

% Inför variabler x0 och x1 som representerar två på varandra
% följande tal i följd
x0=1;x1=(x0+a/x0)/2;

tol=10^-9;
while abs(x0-x1)>tol
    x0=x1;
    x1=(x0+a/x0)/2;
end
end
```

Vi använder sedan funktionen för att skapa tabellen med följande program

```
file='resultat.txt';
fid=fopen( file , 'w' );

% Tabell huvud
fprintf( fid,'          x          sqrt(x)\n');
fprintf( fid,'-----\n');

for i=1:100
    % skriv en rad i tabellen
    r=kvadratrot( i );
    fprintf( fid,'    %3i          %10.8f\n',i,r)
end
fclose(fid)
```