Einführung in die Modellierung

Übung 5

Symbolische/Nummerische Integralrechnung

1. Bestimmen Sie symbolisch die Stammfunktionen folgender Zusammenhänge:

   a) $2x \cdot (x^2+3)$ c) $\frac{\arcsin(x)}{\sqrt{1-x^2}}$

   b) $\frac{1}{x \cdot ln(x^2)}$ d) $\frac{2x^4-3\sqrt{x}}{7\sqrt[3]{x^4}}$

Lösung:

```
clear all
clc

syms x % Definition symbolischer Variablen

f1 = 2*x*(x^2+3); % Definition symbolischer Funktionen
f2 = 1/(x*log(x^2));
f3 = asin(x)/sqrt(1-x^2);
f4 = (2*x^4-3*sqrt(x))/(7*x^(4/3));

I1 = int(f1) % Berechnung der Stammfunktion symbolischer Funktionen
I2 = int(f2)
I3 = int(f3)
I4 = int(f4)
```

```
I1 = (x^2*(x^2 + 6))/2 % Ergebnis
I2 = log(log(x^2))/2
I3 = asin(x)^2/2
I4  = (6*x^(1/6)*(x^(7/2) - 33))/77
```

2. Lösen Sie das folgende bestimmte Integral:

   $\int_0^2 \int_0^1 (2-xy)dxdy$

Lösung:

```
clear all
clc

syms x y % Definition symbolischer Variablen
f = 2 - x*y; % Definition symbolischer Funktion

i1 = int(f,0,2); % Berechnung des bestimmten Integrals über x
i2 = int(i1,y,0,1) % Berechnung des bestimmten Integrals über y
```

```
i2 = 3
```


|Dipl.-Ing. Irina Ikkert, M.Eng.

3. Schreiben Sie eine Funktion *integral*:

function [I] = integral (f,a,b)

mit:
f – Function Handle, zu integrierende Funktion
a – untere Integrationsgrenze
b – obere Integrationsgrenze

Die Funktion soll:
- beim Aufruf mit nur einem Parameter f (Function Handle) das unbestimmte Integral und
- beim Aufruf mit drei Parametern f,a,b das bestimmte Integral der als Function Handle übergebenen Funktion f berechnen und ausgeben.
- beim Aufruf mit zwei Parametern soll das Programm mit einer Fehlermeldung abgebrochen werden.

Testen Sie einige im Matlab vordefinierte und die in der Aufgabe 1 verwendete Funktionen.

Lösung:

```
function [ I ] = integral( f,a,b )
% Die Funktion integral berechnet, in Abhängigkeit von der
%Anzahl der Eingabeparameter, das unbestimmte oder bestimmte
%Integral einer als Function Handle übergebenen Funktion.

syms x      % Definition symbolischer Variablen
f = f(x);   % Definition symbolischer Funktion

 if nargin == 1  I = int(f);  end % unbestimmtes Integral

 if nargin == 3  I = double(int(f,a,b)); end % bestimmtes Int.

 if nargin == 2  error('Ungültige Anzahl der
Eingabeparameter'); end % Abbruchkriterium

end
```

```
>> integral(@(x)(2*x*(x^2+3)),0,1)
ans = 3.5000


>> integral(@sin,0,1)

ans = 0.4597


>> integral(@sin,0)

Error using integral (line 12)

Ungültige Anzahl der Eingabeparameter
```

4. Testen Sie mit Hilfe des bestimmten Integrals die Genauigkeit des Trapez – und Simpsons Verfahrens. Wie groß ist jeweils die Abweichung vom genauen Wert des Integrals?

$$\int_{-2}^{2} (x + 3x \cdot sinx)dx$$

Lösung:

```
clear all
clc

format long % Zahlenformat auf long umstellen
x = -2:.1:2; % Definition des Vektors x für die numerische
Berechnung
y = x+3*x.*sin(x); % Definition der Funktion y für die numerische
Berechnung

I_tr = trapz(x,y)% Berechnung des Integrals nach der Trapezregel

I_sim = quad(@(x)(x+3*x.*sin(x)),min(x),max(x))% Berechnung des
Integrals mit Simpsons-Verfahren

syms x % Definition symbolischer Variablen
y = x+3*x*sin(x); % Definition symbolischer Funktion

I_sym = double(int(y,-2,2)) % symbolische Berechnung des Integrals

format  % Zahlenformat wiederherstellen
delta_tr = I_sym - I_tr % Abweichung für Trapez-Verfahren
delta_sim = I_sym - I_sim % Abweichung für Simpsons-Verfahren
```

```
I_tr  = 10.449933198691181
I_sim = 10.449546600513234
I_sym = 10.449546599519799


delta_tr =  -3.8660e-04

delta_sim = -9.9343e-10
```

➔ Das Simpsons – Verfahren ist genauer!

5. Berechnen Sie symbolisch die Nullstellen und Extrema folgender Funktion und stellen Sie diese mit plot in einem Diagramm grafisch dar

$$y = x^2 \cdot e^{-x^2}$$

Lösung:

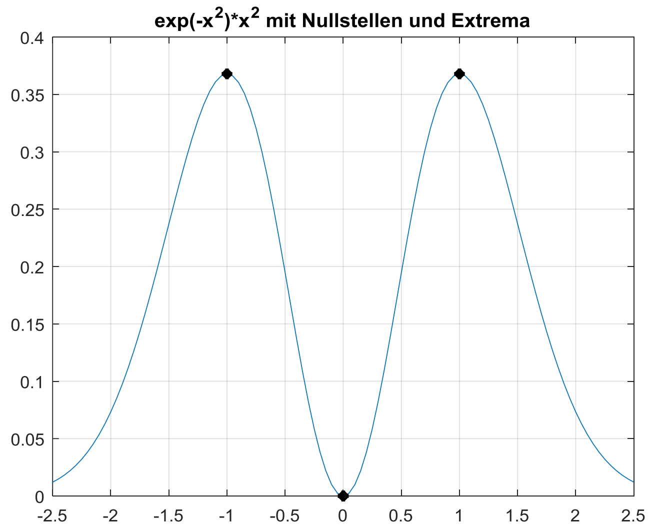
```
syms x % Definition symbolischer Variablen
y = exp(-x^2)*x^2 % Definition symbolischer Funktion

nullx = double(solve(y)); % Berechnung der x Koordinaten der
Nullstellen mit numerischer Umwandlung zu double
nully = double(subs(y,x,nullx)); % Berechnung der Funktionswerte
in den Nullstellen mit numerischer Umwandlung zu double

d1 = diff(y); % Berechnung der ersten Ableitung
nulld1x = double(solve(d1)); % Berechnung der x Koordinaten der
Extrema mit numerischer Umwandlung zu double
nulld1y = double(subs(y,x,nulld1x)); % Berechnung der
Funktionswerte zu den Extrema mit numerischer Umwandlung zu double

% grafische Darstellung der Funktion und ihre Nullstellen + Extrema

x1 = -2.5:0.05:2.5; % Definition des x Vektors
y1 = double(subs(y,x,x1)); % Berechnung der Funktionswerte in x1
plot(x1,y1)
hold on
grid on
plot(nullx,nully,'*k','LineWidth',2,'MarkerSize',5)
hold on
plot(nulld1x,nulld1y,'*k','LineWidth',2,'MarkerSize',5)
title('exp(-x^2)*x^2 mit Nullstellen und Extrema');
```



6. Bestimmen Sie symbolisch die spezielle Lösung der DGL für $y(0) = 2$ und stellen Sie diese im Bereich von -2 bis 8 grafisch dar.

$$y' = -\frac{x \cdot e^{-x}}{3 \cdot y^2}$$

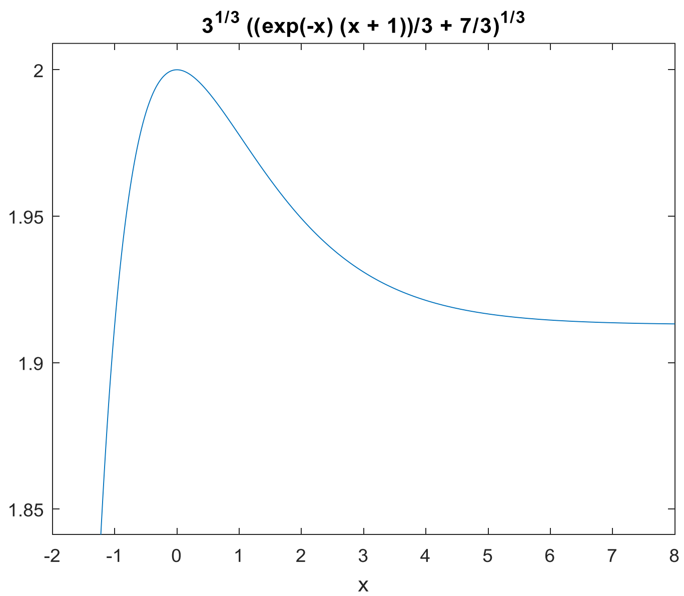
Lösung:

```
clear all
clc

syms x y % Definition symbolischer Variablen

y = dsolve('Dy = -x*exp(-x)/(3*y^2)','y(0)=2','x') % spezielle
Lösung der DGL
ezplot(y,[-2,8]); % grafische Darstellung
```

```
y = 3^(1/3)*((exp(-x)*(x + 1))/3 + 7/3)^(1/3)
```



7. Lösen Sie symbolisch folgende Gleichung mehrerer Variablen nach x auf

$$z = ln\frac{1}{cos^2(x) \cdot cos^2(y)}$$

Lösung:

```
clear all
clc

syms x y % Definition symbolischer Variablen
z = log(1/(cos(x)^2*cos(y)^2)); % Definition symbolischer
Funktion
```

```
z = solve(z) % Lösung der Gleichung nach x
```

```
z =  acos(1/cos(y))
    -acos(1/cos(y))
```

8. Bestimmen Sie symbolisch die Schnittpunkte (xs,ys) folgender Gleichungen und stellen Sie diese mit ezplot in einem Diagramm grafisch dar.

$$y_1 = x^3 - 4x - 5$$

$$y_2 = 3x - 3$$

Lösung:

```
clear all
clc

syms x % Definition symbolischer Variablen
f1 = x^3-4*x-5; % Definition symbolischer Funktion f1
f2 = 3*x-3; % Definition symbolischer Funktion f2

% grafische Darstellung der beiden Funktionen
ezplot(f1,[-3,3])
hold
grid
ezplot(f2,[-3,3])
hold

% symbolische Berechnung der Schnittpunkte mit Umwandlung zu double
xs = double(solve(x^3-4*x-5 == 3*x-3))
ys = double(subs(f1,x,xs))
```

```
xs =

 2.7785
-2.4893
-0.2892


ys  =

5.3354
-10.4679
-3.8675
```