%Numerische Mathematik Serie 1 Marco Weber

%Aufgabe 1

i=eps;

f=0;

while 1+i~=1,

 q=i;

 i=q/2;

 f=f+1;

end

Anzahl=f

eps\_m=i

differenz=eps-i

relativer\_fehler=differenz/eps

%%

%%\_\_\_Lösungen (output)

% Anzahl =

%

% 1

%

%

% eps\_m =

%

% 1.110223024625157e-016

%

%

% differenz =

%

% 1.110223024625157e-016

%

% relativer\_fehler =

%

% 0.500000000000000

%Serie 1 Aufgabe 2 Marco Weber

format long

q=100;

a=0;

%aufsummiert von klein nach gross

for i=2:1:1000

 links=(q+(-1)^(i-1)\*(1/i));

 q=links;

end

n\_links\_rechts=links

%aufsummiert von gross nach klein

for i=2:1:1000

 rechts=a+(-1)^(i-1)\*1/i;

 a=rechts;

end

n\_recht\_links=rechts+100

differenz\_Methoden=n\_links\_rechts-n\_recht\_links

% Aufgrund der Maschinengenauigkeit, ergeben sich beim fortlaugenden

% aufsummieren sich addierende Rundungsfehler. Somit werden kleine Zahlen

% wenn man sie zu grossen Zahlen addiert so gut wie asugelöscht.

%Vergleich mit exaktem Wert

exact=99.69309718305994529172;

%mit methode 1

abs\_fehler\_links\_rechts=exact-n\_links\_rechts

rel\_fehler\_links\_rechts=abs\_fehler\_links\_rechts/n\_links\_rechts

%mit methode 2

abs\_fehler\_rechts\_links=exact-n\_recht\_links

rel\_fehler\_rechts\_links=abs\_fehler\_rechts\_links/n\_recht\_links

%%

%%\_\_\_Lösugen (OUTPUT)

%

% n\_links\_rechts =

%

% 99.692647430560015

%

%

% n\_recht\_links =

%

% 99.692647430559816

%

%

% differenz\_Methoden =

%

% 1.989519660128281e-013

%

%

% abs\_fehler\_links\_rechts =

%

% 4.497524999322877e-004

%

%

% rel\_fehler\_links\_rechts =

%

% 4.511390875094962e-006

%

%

% abs\_fehler\_rechts\_links =

%

% 4.497525001312397e-004

%

%

% rel\_fehler\_rechts\_links =

% 4.511390877090625e-006

%Serie 1 Aufgabe 3 Marco Weber

format long

disp('Teilaufgabe a)') %a)

a=29790;

b=31;

a\_n=(a)^(1/3)-b;

a\_v=(a^2+a\*b+b^2)\*(a-b);

disp('absoluter Fehler:');

a\_f\_abs=a\_v-a\_n

disp('relativer Fehler:');

a\_f\_rel=a\_f\_abs/a\_n

disp('Teilaufgabe b)') %b)

b\_n=log(1001)-log(10000);

b\_v=log(10001/10000);

disp('absoluter Fehler:');

b\_f\_abs=b\_v-b\_n

disp('relativer Fehler:');

b\_f\_rel=b\_f\_abs/b\_n

disp('Teilaufgabe c)') %c)

c\_n=cos(0.00001)-1;

%cos(0,00001)-cos(0)=-2\*sin((x\_1-x\_2)/2)\*sin((x\_1+x\_2)/2)

c\_v=-2\*sin((0.00001+0)/2) \*sin((0.00001-0)/2);

disp('absoluter Fehler:');

c\_f\_abs=c\_v-c\_n

disp('relativer Fehler:');

c\_f\_rel=c\_f\_abs/c\_n

%%

%%\_\_\_\_Lösung (output)

% Teilaufgabe a)

% absoluter Fehler:

%

% a\_f\_abs =

%

% 2.643695970920900e+013

%

% relativer Fehler:

%

% a\_f\_rel =

%

% -7.621690202914462e+016

%

% Teilaufgabe b)

% absoluter Fehler:

%

% b\_f\_abs =

%

% 2.301685587661297

%

% relativer Fehler:

%

% b\_f\_rel =

%

% -1.000043446135852

%

% Teilaufgabe c)

% absoluter Fehler:

%

% c\_f\_abs =

%

% 4.137435214517088e-018

%

% relativer Fehler:

%

% c\_f\_rel =

%

% -8.274869744368383e-008

%Serie 1 Aufgabe 4 Marco Weber

disp('Teilaufgabe a)') %a)

%y=1-exp(x);

%y'=-exp(x);

x=(0.01:0.01:1) ;

%Da Mathlab keine Funktionen darstellen

%kann werden hier einfach 101 Punkte ausgewertet.

xi=abs((-x.\*exp(x))./(1-exp(x)));

%"If" aufgrund der NULL-Division!

%Punkt "." nicht vor den Rec henzeichen Vergessen

%Dadurch wird Elementweise gerechnet!

disp('Maximum der Verstärkerfunktion');

xi\_max=max(xi)

%Dieses Maximum darf den Faktor 10 nicht überschreiten!

%Da 0.1%\*xi<=1% sein muss!

disp('Teilaufgabe b)') %b)

A=hilb(10);

e\_w=eig(A'\*A);

e\_w\_max=max(e\_w);

e\_w\_min=abs(min(e\_w));

disp('Kondition von "hilb"');

kapa=sqrt(e\_w\_max)/sqrt(e\_w\_min)

disp('Kondition von "hilb" mit cond.');

Kondition=cond(A)

disp('Absoluter Fehler');

abs=Kondition-kapa

disp('Relativer Fehler');

rel=abs/kapa

%%

%%\_\_\_Lösungen (output)

%

% Teilaufgabe a)

% Maximum der Verstärkerfunktion

%

% xi\_max =

%

% 1.5820

%

% Teilaufgabe b)

% Kondition von "hilb"

%

% kapa =

%

% 2.7088e+008

%

% Kondition von "hilb" mit cond.

%

% Kondition =

%

% 1.6025e+013

%

% Absoluter Fehler

%

% abs =

%

% 1.6024e+013

%

% Relativer Fehler

%

% rel =

%

% 5.9156e+004

%Serie 1 Aufgabe 5 Marco Weber

A=[0.03482,0.07355,-0.1679;0.07355,0.1554,-0.3548;-0.1679,-0.3548,0.8098];

b=[-1.8656;-3.9423;8.9988];

delta\_b=[1.1172;0.68494;0.53171]\*10^-4;

delta\_x=A^(-1)\*delta\_b;

x=A^(-1)\*b;

disp('relative Fehler von x und b');

rel\_x=norm(delta\_x)/norm(x)

rel\_b=norm(delta\_b)/norm(b)

disp('Verhältnis der rel.-Fehler');

Verhaeltnis=rel\_x/rel\_b

disp('Kondition von A');

Kondition=cond(A)

e\_w\_min=min(eig(A'\*A));

e\_w\_max=max(eig(A'\*A));

Kapa=sqrt(e\_w\_max)/sqrt(e\_w\_min)

%%

%%\_\_\_\_Lösung (output)

% relative Fehler von x und b

%

% rel\_x =

%

% 0.679610559292486

%

%

% rel\_b =

%

% 1.414207299516318e-005

%

% Verhältnis der rel.-Fehler

%

% Verhaeltnis =

%

% 4.805593632029218e+004

%

% Kondition von A

%

% Kondition =

%

% 9.194880161778109e+004

%

%

% Kapa =

%

% 9.194879906271106e+004