Assignment2 MATH 472: Numerical Linear Algebra Kathleen Shannon

The solution to problem 1(2) is: x=
0.0085 0.1575 0.2274 -0.2009

The solution to problem 3(4) is: x=
1.0000 2.0000 3.0000 4.0000

The solution to problem 5 is: x=
1.0000 2.0000 3.0000 4.0000

The solution to problem 6 is: x=
1.0037 1.9925 3.3358 3.6082

the difference between the previous two solutions is
-0.0037 0.0075 -0.3358 0.3918
The difference in the inputs (b values) is:
0.0833

the relative input error would be
0.0435
the relative error would be
-0.0037 0.0037 -0.1007 0.1086

Note that the output error is as much as about 11% more than double the input error of 4.35%

The solution to problem 8 is: x=
1.0000 2.0000 3.0000 4.0000

The solution to problem 9 is: x=
0.7236 2.2902 2.7240 4.2300

the difference between the previous two solutions is
0.2764 -0.2902 0.2760 -0.2300

The difference in the inputs (b values) is:
0.0064

the relative input error would be
0.0056

the relative output error would be
0.2764 -0.1451 0.0920 -0.0575

Note that the output error is as much as about 15% more than 25 times the input error of .56%

The solution to problem 11 is: x=
0.4368 0.1265 0.1838 0.1382 0.1252 0.1107 0.0998 0.0908 0.0826 0.0834

the residual is:
0 0 0 0 0 0 0 0 0 0 ***So I guess that means it is pretty accurate...***

A=

2	1	0	0	0	0	0	0	0	0
1	3	1	0	0	0	0	0	0	0
0	1	4	1	0	0	0	0	0	0
0	0	1	5	1	0	0	0	0	0
0	0	0	1	6	1	0	0	0	0
0	0	0	0	1	7	1	0	0	0
0	0	0	0	0	1	8	1	0	0
0	0	0	0	0	0	1	9	1	0
0	0	0	0	0	0	0	1	10	1
0	0	0	0	0	0	0	0	1	11

Ax= ***all 1's*** residual is ***all 0's*** ***(Matlab printed it in columns and it took up too much space!)***

Using the backslash operator: The solution to problem 11 is: x=
0.4368 0.1265 0.1838 0.1382 0.1252 0.1107 0.0998 0.0908 0.0826 0.0834

Narrative: I actually did the calculations for this before I assigned the problems (and before I checked on initials which is why my matlab filenames do not start KS) First I wrote the tridiag routine and tested it on the example we went over in class. Then I began making the script file to do the problems assigned. I expected that the residual would not be 0 in problem 11 since he asked us to compute it; but it was 0; so I guess the point was that the residual shows that the answer is correct.
Then I added the error check and ran it again. At first I forgot to put error = true in the beginning and I got an error. Once I fixed that everything worked. I ran it earlier today and then block and copied the Command Window into this document so I could comment. I decided to use a different font to distinguish the comments I added from the Matlab output. This was pretty straight forward and aside from typos that I had to fix it went smoothly. I ran the script each time I added another problem so I actually solved problem 2 many many times ☺ My code is below. I used columns on the script to save a page.
Matlab Code:

```
function [ x ] = Tridiag( u,d,l,f,n)
% Tridiag solves a tridiagonal linear system.  The diagonal entries are
% passed in d, the lower diagonal entries in l and the upper diagonal
% entries in u.  The first non-zero entry in l should be l(2).
% f is the right hand side of the equation.
% All of these are 1xn vectors (note this choice is made for readablity,
% u and l actually have only n-1 pieces of information so we could have
% used 1xn-1 vectors.....)
error=false;
for i=2:n
   if abs(d(i-1))<1e-10
       disp('error system is not diagonally dominant system cannot be solved by this routine')
      % if the diagonal entry is too small we do not want to divide by it.
      % This does not actually test for diagonal dominance since that is
      %a sufficient but not necessary condition for solvability.
       error = true; %indication that the system hasn't been solved.
   else
       d(i)=d(i)-l(i)*u(i-1)/d(i-1);
       f(i)=f(i)-l(i)*f(i-1)/d(i-1);
   end
end
if not(error)
   x(n)=f(n)/d(n);
   for i=n-1:-1:1
       x(i)=(f(i)-u(i)*x(i+1))/d(i);
   end
else
   for i=n-1:-1:1
       x(i)=9999;
   end
end
end
```

```
%This script sets up the values for the problems in Assignment 2, calls the
%tridiag routine (I did not call it KS tridiag - I should have) and does
%other calculations to investigate the questions asked.
%Problem 2
 l1=[0,1,1,2];
d1=[4,4,4,4];
u1=[2,1,1,0];
f1=[pi/9, sqrt(3)/2,sqrt(3)/2,-pi/9];
x1=Tridiag( u1,d1,l1,f1,4);
disp('The solution to problem 1 is: x=')
disp(x1)
%Problem 4:
d2=[6,4,4,6];
l2=[0,2,1,1];
u2=[1,1,2,0];
f2=[8,13,22,27];
x2=Tridiag( u2,d2,l2,f2,4);
disp('The solution to problem 3 is: x=')
disp(x2)
%Problem 7 - 5
u3=[1/2,1/4,1/6,0];
d3=[1,1/3,1/5,1/7];
l3=[0,1/2,1/4,1/6];
f3=[2,23/12,53/30,15/14];
x3=Tridiag( u3,d3,l3,f3,4);
disp('The solution to problem 5 is: x=')
disp(x3)
%Problem 7 -6:
f4=[2,2,53/30,15/14];
x4=Tridiag( u3,d3,l3,f4,4);
disp('The solution to problem 6 is: x=')
disp(x4)
e34=x3-x4;
disp('the difference between the previous two solutions is')
disp(e34)
e34=e34./x4;
disp('The difference in the inputs (b values) is:')
inerror=abs(f3(2)-f4(2));
disp(inerror)
disp('the relative input error would be')
inerror=inerror/f3(2);
disp(inerror)
disp('the relative error would be')
disp(e34);
%Problem 10-8
u5=[10/21,1/13,1/21];
d5=[1/2,1/3,1/4,1/5];
l5=[0,1/4,1/5,1/6];
f5=[61/42,179/156,563/420,13/10];
x5=Tridiag(u5,d5,l5,f5,4);
disp('The solution to problem 8 is: x=')
disp(x5)
%Problem 10-9
f6=[61/42,180/156,563/420,13/10];
x6=Tridiag(u5,d5,l5,f6,4);
disp('The solution to problem 9 is: x=')
disp(x6)
e89=x5-x6;
disp('the difference between the previous two solutions is')
disp(e89)
disp('The difference in the inputs (b values) is:')
inerror=abs(f5(2)-f6(2));
disp(inerror)
e89=e89./x5;
disp('the relative input error would be')
inerror=inerror/f5(2);
disp(inerror)
disp('the relative output error would be')
disp(e89);
d7=zeros(1,10);
u7=zeros(1,10);
l7=zeros(1,10);
f7=zeros(1,10);
for i=1:10
    d7(i)=i+1;
    u7(i)=1;
    l7(i)=1;
    f7(i)=1;
end
x7=Tridiag(u7,d7,l7,f7,10);
disp('The solution to problem 11 is: x=')
disp(x7)
r=zeros(1,10);
for i=2:9
    r(i)=x7(i-1)+(i+1)*x7(i)+x7(i+1)-1;
end
r(1)=2*x7(1)+x7(2)-1;
r(10)=11*x7(10)+x7(9)-1;
disp('the residual is:')
disp(r)
A=zeros(10,10);
A(1,1)=2;
A(1,2)=1;
A(10,9)=1;
A(10,10)=11;
f7(1)=1;
f7(10)=1;
for i=2:9
    A(i,i)=i+1;
    A(i,i-1)=1;
    A(i,i+1)=1;
    f7(i)=1;
end
b=A*x7';
disp('A=')
disp(A)
disp('Ax=')
disp(b)
r=b-f7';
disp('residual is')
disp(r)
x=A\b;
disp('The solution to problem 11 is: x=')
disp(x')
```