```
»%Here are the examples from class, Monday, Jul 19
»%First, the code for the solution of a tridiagonal system:
»type tridag

function w=tridag(a,b,c,f)
  m=length(a);

  alph=[a(1)]; gam=[c(1)/a(1)];
   j=2;
    while j<=m-1
     alph=[alph,a(j)-b(j)*gam(j-1)];
     gam=[gam,c(j)/alph(j)];
      j=j+1;
    end
  alph=[alph,a(m)-b(m)*gam(m-1)];

  j=2;
   z=[f(1)/alph(1)];
    while j<=m
     z=[z,(f(j)-b(j)*z(j-1))/alph(j)];
      j=j+1;
    end

   w=zeros(m,1);w(m)=z(m);
   j=m-1;
    while j>=1
     w(j)=z(j)-gam(j)*w(j+1);
      j=j-1;
    end


»%Now for an example:
»a

a =

     4     4     4     4     4     4     4     4     4     4

»b

b =

     0     1     1     1     1     1     1     1     1     1
```

```
»c

c =

     1     1     1     1     1     1     1     1     1     0

»f

f =

     1     2     3     4     5     6     7     8     9    10

»%Let's count operations.
»flops(0)
»tridag(a,b,c,f)

ans =

    0.1667
    0.3333
    0.5000
    0.6668
    0.8327
    1.0025
    1.1572
    1.3686
    1.3684
    2.1579

»n1=flops

n1 =

   136

»%Let's compare the efficiency of this scheme with plain-ole Gauss:
»A=[4 1 0 0 0 0 0 0 0 0;1 4 1 0 0 0 0 0 0 0;0 1 4 1 0 0 0 0 0 0;
    0 0 1 4 1 0 0 0 0 0;0 0 0 1 4 1 0 0 0 0;0 0 0 0 1 4 1 0 0 0;
    0 0 0 0 0 1 4 1 0 0;0 0 0 0 0 0 1 4 1 0;0 0 0 0 0 0 0 1 4 1;
    0 0 0 0 0 0 0 0 1 4]

A =
```

```
     4     1     0     0     0     0     0     0     0     0
     1     4     1     0     0     0     0     0     0     0
     0     1     4     1     0     0     0     0     0     0
     0     0     1     4     1     0     0     0     0     0
     0     0     0     1     4     1     0     0     0     0
     0     0     0     0     1     4     1     0     0     0
     0     0     0     0     0     1     4     1     0     0
     0     0     0     0     0     0     1     4     1     0
     0     0     0     0     0     0     0     1     4     1
     0     0     0     0     0     0     0     0     1     4

»flops(0)
»A\f'

ans =

    0.1667
    0.3333
    0.5000
    0.6668
    0.8327
    1.0025
    1.1572
    1.3686
    1.3684
    2.1579

»n2=flops

n2 =

   785

»%Finally, let's see how terrible matrix inversion would be.
»flops(0)
»inv(A)*f'

ans =

    0.1667
    0.3333
    0.5000
```

```
    0.6668
    0.8327
    1.0025
    1.1572
    1.3686
    1.3684
    2.1579

»n3=flops

n3 =

        2049

»%Look at the operation counts again:
»[n1 n2 n3]

ans =

         136         785        2049

»
```