

More on Gauss-Seidel

Here is the Matlab code:

```
function y=GaussSeidel(A,b,x,NumIters)
% Runs the Gauss-Seidel method for solving Ax=b, starting with x and
% running a maximum of NumIters iterations.
%
% The matrix A should be diagonally dominant, and in particular, it should
% not have any diagonal elements that are zero (a division by zero error
% will be produced).
%
% The output y will be the whole sequence of outputs instead of the final
% value (if x is in R^n, then y will be n x NumIters

D=diag(A);
A=A-diag(D);
D=1./D; %We need the inverses

n=length(x);
x=x(:); %Make sure x is a column vector
y=zeros(n,NumIters);

for j=1:NumIters
    for k=1:n
        x(k)=(b(k)-A(k,:)*x)*D(k);
    end
    y(:,j)=x;
end
```

Here is some sample output for iterating on Example 2.22 of our text:

```
>> A=[3,1,-1;2,4,1;-1,2,5]; b=[4;1;1];
>> x0=[0;0;0];
>> y=GaussSeidel(A,b,x0,7)
y =
    1.3333    1.6833    1.8622    1.9396    1.9735    1.9884    1.9949
   -0.4167   -0.7500   -0.8903   -0.9519   -0.9789   -0.9908   -0.9960
    0.6333    0.8367    0.9286    0.9687    0.9863    0.9940    0.9974
>>
```