

```
%Gaussian elimination for a structured matrix
%Devise an efficient way to arrange the computations for solving an n-by-n
%linear system with non-zero entries in the coefficient matrix only in the
%first and last rows and columns and also in the two main diagonals.
```

```
clear all
close all
```

```
m = 9; n = 9;
```

```
%Sample matrix A, to check the algorithm
```

```
A = matrix(m,n);
```

```
%seed
```

```
rng('default')
```

```
s = rng
```

```
b = randn(n,1);
```

```
x = A\b
```

```
[a,x] = guas(A,b)
```

```
%Gauss elimination
```

```
function [a,x] = guas(A,b)
```

```
    a = [A,b];
```

```
    [n,m] = size(a);
```

```
    %forward sub
```

```
    for i = 2:n-1
```

```
        if n-i+1 ~= i
```

```
            temp = a(i,:)*a(n-i+1,i);
```

```
            a(n-i+1,:) = a(n-i+1,:) - temp/a(i,i);
```

```
        %else
```

```
            % break
```

```
        end
```

```
    end
```

```
    for j = 1:n-1
```

```
        a(n,:) = a(n,:) - a(j,:)*a(n,j)/a(j,j);
```

```
    end
```

```
    %swap first row with the last one
```

```
    temp = a(1,:);
```

```
    a(1,:) = a(n,:);
```

```
    a(n,:) = temp;
```

```
    for j = 2:n
```

```
        a(j,:) = a(j,:) - a(1,:)*a(j,1)/a(1,1);
```

```
    end
```

```
    for j = 2:n-1
```

```
        a(n,:) = a(n,:) - a(j,:)*a(n,j)/a(j,j);
```

```
    end
```

```
    %back sub
```

```
    x = zeros(n,1);
```

```
    x(n) = a(n,m)/a(n,n);
```

```

    for i = n-1:-1:1
        temp = a(i,n)*x(n);
        x(i) = (a(i,m) - temp)/a(i,i);
    end
end

%unstructured matrix
function A = matrix(m,n)
    rng('default')
    s = rng
    A = diag(randn(n,1));
    A = fliplr(A);
    for j = 1:m
        for i = 1:n
            if i == j
                A(j,i) = randn(1);
            elseif i == 1
                A(j,i) = randn(1);
            elseif j == 1
                A(j,i) = randn(1);
            elseif i == n
                A(j,i) = randn(1);
            elseif j == n
                A(j,i) = randn(1);
            end
        end
    end
end
end
end

```

s =

struct with fields:

```

    Type: 'twister'
    Seed: 0
    State: [625x1 uint32]

```

s =

struct with fields:

```

    Type: 'twister'
    Seed: 0
    State: [625x1 uint32]

```

x =

```

    0.2973
   -5.7976
   -3.5566
   -3.4111
   11.4735
    1.7523
    2.4278

```

3.8738
3.7633

a =

Columns 1 through 7

-26.0704	0	0	0	0	0	0
0	1.4172	0	0	0	0	0
0	0	0.7172	0	0	0	0
0	0	0	1.0347	0	0	0
0	0	0	0	0.2939	0	0
0	0	0	0	0	-0.0574	0
0	0	0	0	0	0	-4.3098
0.0000	0	0	0	0	0	0
-0.0000	0	0	0	0	0	0

Columns 8 through 10

0	5.0366	11.2032
0	2.7878	2.2750
0	0.2824	-1.4881
0	2.9355	7.5177
0	-0.8459	0.1884
0	0.1408	0.4293
0	2.1265	-2.4607
-1.1983	1.2050	-0.1074
0	2.7722	10.4327

x =

0.2973
-5.7976
-3.5566
-3.4111
11.4735
1.7523
2.4278
3.8738
3.7633

.....