

## 2.2.2. Implementação do Método de Gauss

Seguem, abaixo, a implementação do método pela sub-rotina GAUSS e um exemplo de programa para usá-la.

```
Borland Pascal - [c:\pendrive\icci\sisgauss.pas]
File Edit Search Run Compile Tools Options Window Help
? [Icons]

program sisgauss;
uses wincrt;
var
  a,m : array[1..50,1..50] of real;
  {Cálculo para sistemas até 51 linhas e 51 colunas
   b contém os coeficientes e x são as variáveis }

  n : integer; { Número lido no teclado para o cálculo}

  x,b : array[1..50] of real; { Variáveis procuradas }

  L,C,k,i,j: integer; { Contadores do Loop }
  s : real;

function potencia(x :real; y:integer):real;
var
  n : integer;    h : real;
begin
  h := x;
  for n:= 1 to y do h:= h*x;
  potencia := h;
end;

begin
  writeln('Solução de sistemas triangulares');
  writeln; write('Digite o número de linhas ');
  readln(n);
  for L := 1 to n do begin
    for C := 1 to n do begin
      write('Posição [' ,L,',',C,'] = '); readln(a[L,C]);
    end;
    write('Termo independente [' ,L,',',n+1,'] = '); readln(b[L]);
  end;
end;
```

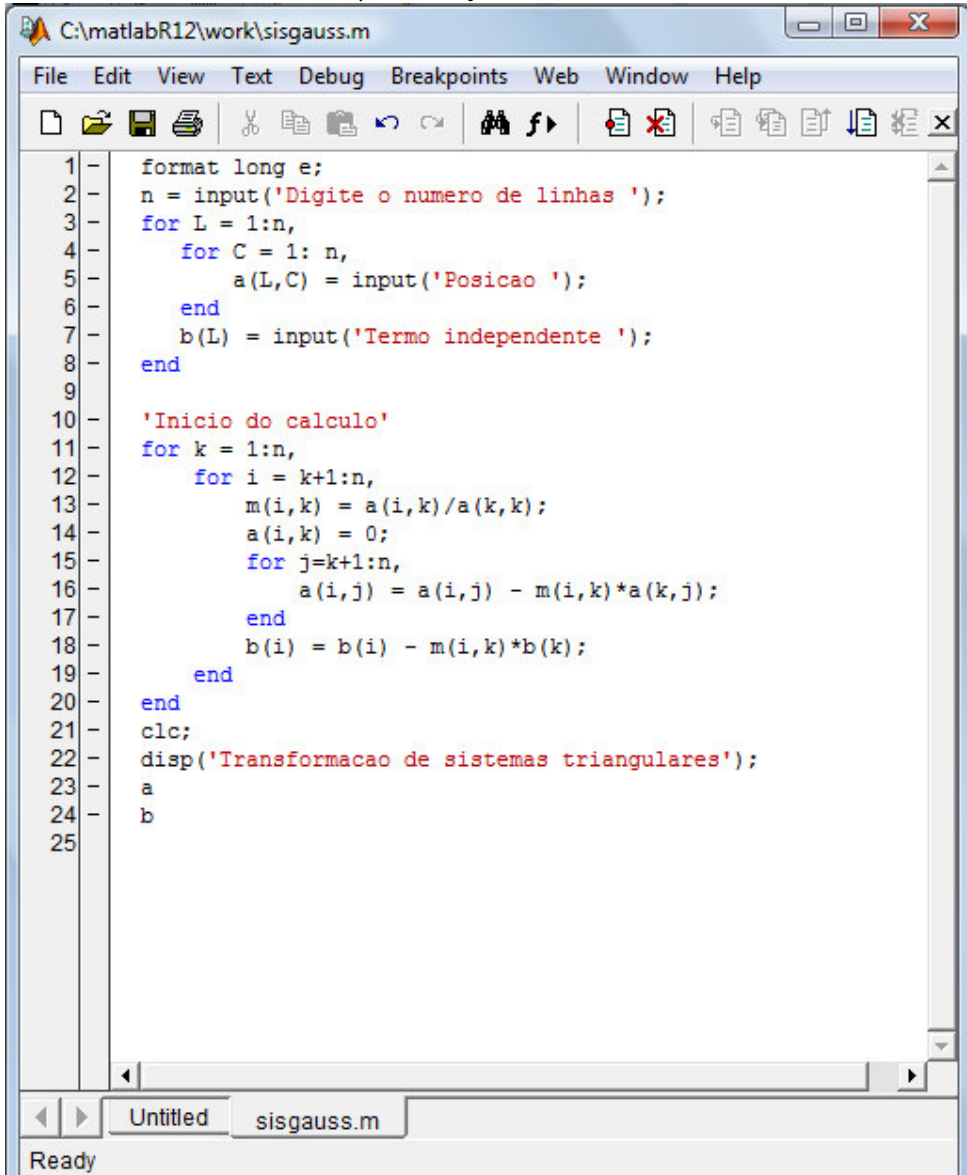
Os dois módulos compõe um mesmo programa que transforma um sistema de n equações e n variáveis em sistema triangular superior.

```
Borland Pascal - [c:\pendrive\icci\sisgauss.pas]
File Edit Search Run Compile Tools Options Window Help
? [Icons]

{ Início do cálculo }

for k := 1 to n do begin
  for i := k + 1 to n do begin
    m[i,k] := a[i,k]/a[k,k];
    a[i,k] := 0;
    for j := k + 1 to n do begin
      a[i,j] := a[i,j] - m[i,k]*a[k,j];
    end;
    b[i] := b[i] - m[i,k]*b[k];
  end;
end;
clrscr;
writeln('Transformação em sistemas triangulares');
writeln;
for l := 1 to n do begin
  for c := 1 to n do begin
    write(a[l,c], ' ');
  end;
  writeln(' = ',b[l]);
end;
end.
```

O módulo abaixo é a mesma implementação no MatLab.



```
1 - format long e;  
2 - n = input('Digite o numero de linhas ');  
3 - for L = 1:n,  
4 -     for C = 1: n,  
5 -         a(L,C) = input('Posicao ');  
6 -     end  
7 -     b(L) = input('Termo independente ');  
8 - end  
9 -  
10 - 'Inicio do calculo'  
11 - for k = 1:n,  
12 -     for i = k+1:n,  
13 -         m(i,k) = a(i,k)/a(k,k);  
14 -         a(i,k) = 0;  
15 -         for j=k+1:n,  
16 -             a(i,j) = a(i,j) - m(i,k)*a(k,j);  
17 -         end  
18 -         b(i) = b(i) - m(i,k)*b(k);  
19 -     end  
20 - end  
21 - clc;  
22 - disp('Transformacao de sistemas triangulares');  
23 - a  
24 - b  
25 -
```

Os dois exemplos de código acima implementam a transformação de um sistema  $N \times N$  em um sistema de matriz triangular superior. Se combinarmos o programa SISGAUSS com o SISTRI resolveremos sistemas com até 50 variáveis (em pascal) ou até 100 (com MatLab).