x = gauss(S,f)

```
function x = gauss(S,f)
n = length(f);
S = [S | f]        Augment S with f
for k=1:n-1        k counts columns
  1. r = row number, larger than or equal to k,
       with largest value (in magnitude) in column k
  2. if this largest value is really small then warn the user
  3. swap row r and row k  ⟶  S([j k],:) = S([k j],:)
     for j=k+1:n
       mix row k into row j in order to eliminate S(j,k)
     end
end
1. if S(n,n) is really small then warn the user
2. strip off the changed f, i.e., copy column n+1 of S onto f
3. x = trisolve(S,f)
return
```

< eps

★ 1. r = row number, larger than or equal to k,
with largest value (in magnitude) in column k

If A is an n-by-n matrix and I look down column k and ask for the row number, at or below the diagonal, that contains the largest element, then I should type which one of the following choices to get the correct row number: rnum?

A. [y, rnum] = max(abs(A(k:n,k)))

B. [y,rnum] = max(abs(A(k:k:n)))

C. [y, rnum] = max(abs(A(k:n,k));
rnum = rnum + k – 1;

★ 1. r = row number, larger than or equal to k,
with largest value (in magnitude) in column k

```
Saug =

     4     2     1     1
     2     4     2     2
     1     2     4     3

>> k

k =

     2

>> n

n =

     3

>> [y,i] = max(abs(Saug(k:n,k)))

y =

     4


i =

     1
```

```
Saug =

     4     2     1     1
     2     4     2     2
     1     2     4     3
```

Row 1 *of the submatrix we are considering!!*

★ mix row k into row j in order to eliminate S(j,k)

```
for j=k+1:n
   S(j,:) = S(j,:) + magicnumber*S(k,:)
end
```

before the following line, we need to calculate this magic number. What is it? (I claim it is a one-liner)

row_j (where we are trying to eliminate something) = row_j + (some multiple, m, of) row_k

$$\begin{matrix} \textcircled{4} & 2 & 1 & 1 \\ \textcircled{1} & 2 & 4 & 3 \\ 2 & 4 & 2 & 2 \end{matrix} \xrightarrow{\text{m}} \text{row}_2 = \text{row}_2 + (-1/4)\text{row}_1 \xrightarrow{} \begin{matrix} 4 & 2 & 1 & 1 \\ 0 & 3/2 & 15/4 & 11/4 \\ 0 & 3 & 3/2 & 3/2 \end{matrix}$$

k = 1
j = 2

x = gauss(S,f)

```
function x = gauss(S,f)
n = length(f);
★ S = [S | f]      Augment S with f
for k=1:n-1        k counts columns
★   1. r = row number, larger than or equal to k,
          with largest value (in magnitude) in column k
    2. if this largest value is really small then warn the user
★   3. swap row r and row k
       for j=k+1:n
         ★ mix row k into row j in order to eliminate S(j,k)
       end
end
1. if S(n,n) is really small then warn the user
2. strip off the changed f, i.e., copy column n+1 of S onto f
3. x = trisolve(S,f)
return
```

< eps

x = trisolve(S,f)

Backsubstitution code given in notes! Observe that S must be triangular.

```
x = zeros(n,1);
x(n) = f(n)/S(n,n);
for j=n-1:-1:1
    tmp = 0;
    for k=j+1:n
        tmp = tmp + S(j,k)*x(k);
    end
    x(j) = (f(j) - tmp)/S(j,j);
end
```