```
julia> A=[1 2 3; 4 5 6; 7 8 9]
3×3 Matrix{Int64}:
 1  2  3
 4  5  6
 7  8  9

julia> using LinearAlgebra

julia> det(A)
0.0
```

complains if A is not invertible...

```
julia> lu(A)
ERROR: SingularException(3)
Stacktrace:
 [1] checknonsingular
   @ /buildworker/worker/package_l
tdlib/v1.6/LinearAlgebra/src/facto
 [2] checknonsingular
   @ /buildworker/worker/package_l
tdlib/v1.6/LinearAlgebra/src/facto
 [3] #lu!#136
   @ /buildworker/worker/package_l
tdlib/v1.6/LinearAlgebra/src/lu.jl
 [4] #lu#140
   @ /buildworker/worker/package_l
tdlib/v1.6/LinearAlgebra/src/lu.jl
 [5] lu (repeats 2 times)
   @ /buildworker/worker/package_l
tdlib/v1.6/LinearAlgebra/src/lu.jl
 [6] top-level scope
   @ REPL[4]:1
```

encourage Julia to factor a non-invertible matrix

```
julia> lu(A,check=false)
Failed factorization of type LU{Float64, Matrix{Float64}}

julia> L,U,P=lu(A,check=false)
Failed factorization of type LU{Float64, Matrix{Float64}}

julia> L
3×3 Matrix{Float64}:
 1.0       0.0  0.0
 0.142857  1.0  0.0
 0.571429  0.5  1.0

julia> U
3×3 Matrix{Float64}:
 7.0  8.0       9.0
 0.0  0.857143  1.71429
 0.0  0.0       0.0
```

some pivoting happened to reduce rounding error...
Chose largest pivot

```
julia> L*U
3×3 Matrix{Float64}:
 7.0  8.0  9.0
 1.0  2.0  3.0
 4.0  5.0  6.0
```

P tells the row swaps...

```
julia> P
3-element Vector{Int64}:
 3
 1
 2
```

Householder reflectors from last time: $H = I - 2vv^T$ where $\|v\|=1$

$\|a_1\|^2 = c^2$ $\qquad (I-2vv^T)a_1 = ce_1$

Now solve for $v$: $\qquad a_1 - 2v(v^Ta_1) = ce_1$

$$v = \frac{a_1 - ce_1}{2v^Ta_1} = \frac{a_1 - ce_1}{\|a_1 - ce_1\|}$$ since it's a unit vector..

Let's find a reflector so that $H\begin{bmatrix}1&2&3\\4&5&6\\7&8&9\end{bmatrix} = \begin{bmatrix}c & ? \\ 0 & \\ 0 & \cdot\end{bmatrix}$

$a_1$ is the first column of A.

$$c = \pm\|a_1\| = \pm\left\|\begin{bmatrix}1\\4\\7\end{bmatrix}\right\|$$

$$= \pm\sqrt{1+16+49} \approx 8.1240\ldots$$

```
julia> A
3×3 Matrix{Int64}:
 1  2  3
 4  5  6
 7  8  9

julia> norm(A[:,1])
8.12403840463596
```

$$v = \frac{a_1 - ce_1}{\|a_1 - ce_1\|} \approx \begin{bmatrix}0.749\ldots\\0.328\ldots\\0.575\ldots\end{bmatrix}$$

choose $\pm$ to make denominator big...

```
julia> e1=[1,0,0]
3-element Vector{Int64}:
 1
 0
 0

julia> c=norm(A[:,1])
8.12403840463596

julia> norm(A[:,1]-c*e1)
10.758806773556634

julia> norm(A[:,1]+c*e1)
12.175716685652304
```

the bigger one

```
julia> v1=(A[:,1]+c*e1)/norm(A[:,1]+c*e1)
3-element Vector{Float64}:
 0.7493635602894408
 0.32852275584841295
 0.5749148227347227
```

$$H_1 = I - 2vv^T$$

The Householder reflector

```
julia> H1=I-2*v1*v1'
3×3 Matrix{Float64}:
 -0.123091  -0.492366  -0.86164
 -0.492366   0.784146  -0.377745
 -0.86164   -0.377745   0.338946
```

```
julia> H1*A
3×3 Matrix{Float64}:
 -8.12404       -9.60114    -11.0782
  8.88178e-16  -0.0859656    -0.171931
  8.88178e-16  -0.90044      -1.80088
```

essentiall 0 up to rounding...

focus on this submatrix to finish the QR factorization...

```
julia> A2=(H1*A)[2:3,2:3]
2×2 Matrix{Float64}:
 -0.0859656  -0.171931
 -0.90044    -1.80088
```

now find another reflector to make this in the form $\begin{bmatrix} c_2 & ? \\ 0 & \end{bmatrix}$

$$c_2 = \left\| \begin{bmatrix} -0.0859656 \\ -0.90044 \end{bmatrix} \right\| = 0.945534\ldots$$

```
julia> norm(A2[:,1])
0.9045340337332901
```

but for the submatrix

$$v_2 = \frac{a_1 - c e_1}{\|a_1 - c e_1\|} = \begin{bmatrix} -0.7399\ldots \\ -0.6726\ldots \end{bmatrix}$$

```
julia> c2=norm(A2[:,1])
0.9045340337332901

julia> e1=[1,0]
2-element Vector{Int64}:
 1
 0

julia> norm(A2[:,1]-c2*e1)
1.3386116703491497

julia> norm(A2[:,1]+c2*e1)
1.21690018848397 2
```

```
julia> v2=(A2[:,1]-c2*e1)/norm(A2[:,1]-c2*e1)
2-element Vector{Float64}:
 -0.7399454417565073
 -0.6726668887523506
```

Now to go back to the original size matrix

$$\tilde{v}_2 = \begin{bmatrix} 0 \\ -.07399\ldots \\ -0.6726\ldots \end{bmatrix}$$

```
julia> v2o=[0; v2]
3-element Vector{Float64}:
  0.0
 -0.7399454417565073
 -0.6726668887523506
```

```
julia> H2=I-2*v2o*v2o'
3×3 Matrix{Float64}:
 1.0   0.0         0.0
 0.0  -0.0950385  -0.995474
 0.0  -0.995474    0.0950385
```

good since the first column is already okay...

```
julia> H2*H1*A
3×3 Matrix{Float64}:
 -8.12404      -9.60114     -11.0782
 -8.88178e-16   0.904534      1.80907
 -8.88178e-16  -4.44089e-16   8.88178e-16
```

zero up to rounding

this zero because the A was singular...

$$H_2 H_1 A = R$$

$$A = H_1^{-1} H_2^{-1} R = H_1^T H_2^T R = H_1 H_2 R = QR$$

where $Q = H_1 H_2$.