routh(coeffs, eps)

The function takes an array of coefficients representing a characteristic equation and the number eps, which may replace zeros.

For example, obtain the characteristic equation of a system, then compute its Routh-array for stability analysis.

```
clear; syms s K_p K_i K;

sys1 = K/(s*(s+1)*(s+2)) + 1; % Unity gain system.

sys2 = 1 + K_p*(2/(s^3 + 4*s^2 + 5*s + 2)); % P-controlled system.

K_p = 3; % Keeping one parameter constant.
sys3 = 1 + (K_p + K_i/s)*(2/(s^3 + 4*s^2 + 5*s + 2)); % PI-controlled system.

coeffs = ELAB.sd2ce(sys3)
```

coeffs =

$$\begin{pmatrix} 1 & 4 & 5 & 8 & 2K_i \end{pmatrix}$$

```
RA = ELAB.routh(coeffs)
```

RA =

$$\begin{pmatrix} 1 & 5 & 2K_i \\ 4 & 8 & 0 \\ 3 & 2K_i & 0 \\ 8-\frac{8K_i}{3} & 0 & 0 \\ 2K_i & 0 & 0 \end{pmatrix}$$

critical(RA)

Takes a Routh-array and attempts to find critical values, important for stability analysis.

```
ELAB.critical(RA, true)
```

Solving equations for param:

$$8-\frac{8K_i}{3}=0$$

$$2K_i=0$$

For stability, param (K) =]3;0[

From s^2 auxiliary equation:

$$\begin{pmatrix} 3s^2+3=0 & 3s^2=0 \end{pmatrix}$$

Solved for s:

$$\begin{pmatrix} -\mathrm{i} & 0 \\ \mathrm{i} & 0 \end{pmatrix}$$