Euler scheme for systems of ODEs

$$(IVP)\begin{cases} \vec{y}\,' = \vec{F}(t,\vec{y}), & t_0 < t \leq t_{end} \\ \vec{y}(t_0) = \vec{y}_0 \end{cases}$$

Discretize time: $t_n = t_0 + n\cdot\Delta t$, $\Delta t = \dfrac{t_{end} - t_0}{N_{steps}}$, $\vec{Y}^n \approx \vec{y}(t_n)$

Euler: $\vec{Y}^0 = \vec{y}_0$

$\vec{Y}^{n+1} = \vec{Y}^n + \Delta t \cdot \vec{F}(t_n, \vec{Y}^n)$, $n = 0, 1, \dots, N$

It is a first order scheme: error $= O(\Delta t)$

e.g. for 2 ODEs: $\begin{cases} y_1' = F_1(t, y_1, y_2), & y_1(t_0) = y_1^0 \\ y_2' = F_2(t, y_1, y_2), & y_2(t_0) = y_2^0 \end{cases}$

it is simpler to code each component:

$$Y_1^0 = y_1^0, \quad Y_1^{n+1} = Y_1^n + \Delta t \cdot F_1(t_n, Y_1^n, Y_2^n)$$
$$Y_2^0 = y_2^0, \quad Y_2^{n+1} = Y_2^n + \Delta t \cdot F_2(t_n, Y_1^n, Y_2^n). \qquad n=0,1,\dots,N$$

Both F_1 and F_2 should be evaluated in a FCN subprogram:

```
function [F1n, F2n] = FCN(tn , Y1n , Y2n)
    F1n = formula for F1(tn, Y1n, Y2n);
    F2n =    "     "  F2(             );
end
```

Note the great advantage of explicit vs implicit schemes.

In implicit scheme would need to solve the coupled system, at each time step, for the components Y_1^{n+1}, Y_2^{n+1}, using some system solver for nonlinear equations!

e.g. Newton-Raphson for systems, or fixed-point iterations (Gauss-Seidel), or more advanced (there are many...).