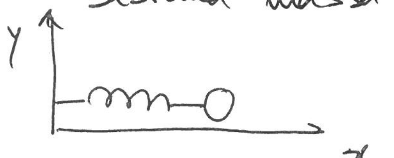
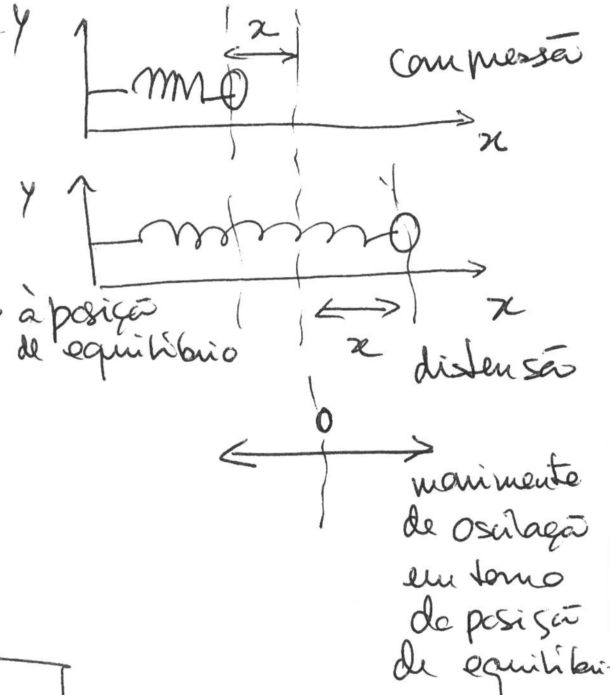
Nota de apoio ao problema **2.1** (Trabalho Prático 2)

①

Aplicação dos Métodos de Euler Implícito e Crank-Nicolson à equação do movimento do Oscilador Harmónico Simples (OHS)

Sistema massa-mola





$x \rightarrow$ deslocamento da massa em relação à posição de equilíbrio

$$\boxed{F = -Kx} \Rightarrow \quad (1)$$

$$m\,a_x = -Kx \Leftrightarrow$$

$$m \frac{d^2x}{dt^2} = -Kx$$

$$\Leftrightarrow \boxed{a_x = -\frac{K}{m} x}$$

e $\boxed{\frac{K}{m} = w^2}$

$$\Leftrightarrow \begin{cases} \dfrac{dv_x}{dt} = -\dfrac{K}{m} x \\[2ex] \dfrac{dx}{dt} = v_x \end{cases} \quad (2)$$

Discretização da equação 2º o método de Euler-Implícito (EI)

Para uma equação do tipo $\frac{dy}{dt} = f(t, y)$ o algoritmo de (EI), escreve-se como:

$$\boxed{y_{k+1} = y_k + f(t_{k+1}, y_{k+1}) \times h}$$

aplicando-o a cada equação do sistema (2), tem-se:

$$\begin{cases} v_{x_{k+1}} = v_{x_k} + \left(-\frac{K}{m} x_{k+1}\right) \times h & (1A) \\[2ex] x_{k+1} = x_k + v_{x_{k+1}} \times h & (1B) \end{cases}$$

O sistema pode ser resolvido nesta forma? (Programado) Porquê?

Se substituirmos $v_{x_{k+1}}$, eq. (1A), na equação (1B) e se resolvermos em ordem a x_{k+1}, obtemos:

$$\begin{cases} x_{k+1} = \dfrac{x_k + vx_k \cdot h}{1 + w^2 h^2} \\ vx_{k+1} = vx_k - w^2 x_{k+1} \cdot h \end{cases}$$

1) E este sistema, já pode ser programado? Porquê? Use um **ciclo FOR** para o fazer.

2) <u>Função</u> <u>**LINSOLVE**</u> do <u>Matlab</u>

Podemos usar a função intrínseca do Matlab. Para tal é necessário escrever o sistema na forma matricial

$$\boxed{AX = b}$$

Se partirmos do sistema de equações (1.A) e (1.B), podemos reescrever este sistema isolando no membro esquerdo os termos c/ índice k+1 e no membro direito os termos com índice k, (2)

$$\begin{cases} \underline{x_{k+1}} + (-h)\,\underline{vx_{k+1}} = \underline{x_k} \\ w^2 h\, \underline{x_{k+1}} + v\cancel{x}_{k+1} = \underline{vx_k} \end{cases}$$

na forma matricial, tem-se

$$\underbrace{\overbrace{\begin{bmatrix} 1 & -h \\ w^2 h & 1 \end{bmatrix}}^{A}}_{\text{A matriz constante}} \underbrace{\overbrace{\begin{bmatrix} x_{k+1} \\ vx_{k+1} \end{bmatrix}}^{X}}_{\text{vetor que pretendemos calcular}} = \underbrace{\overbrace{\begin{bmatrix} x_k \\ vx_k \end{bmatrix}}^{b}}_{\text{vetor de termos independentes que varia de iteração para iteração}}$$

<u>Em Matlab:</u> etc. (definir os parâmetros antes)

A=[1 -h; w^2*h 1];

for . . .

Discretização da equação 2º o método de Crank-Nicolson (CN)

Para $\frac{dy}{dt} = f(t, y)$, tem-se usando (CN)

$$\boxed{y_{k+1} = y_k + \frac{1}{2}\left[f(t_k, y_k) + f(t_{k+1}, y_{k+1})\right] * h}$$

Aplicando este algoritmo a cada equação do Sistema (2), tem-se:

$$\begin{cases} vx_{k+1} = vx_k - w^2(x_{k+1} + x_k) * \frac{h}{2} & (2A) \\ x_{k+1} = x_k + (vx_{k+1} + vx_k) * \frac{h}{2} & (2B) \end{cases}$$

Substituindo vx_{k+1}, (eq (2A)), na equação (2B) e se resolvermos em ordem a x_{k+1}, obtém-se:

$$\begin{cases} x_{k+1} = \dfrac{x_k(1 - w^2 h^2/4) + vx_k * h}{(1 + w^2 h^2/4)} \\ vx_{k+1} = vx_k - w^2(x_{k+1} + x_k) * h/2 \end{cases}$$

Matlab

```
H = ____

for k=1:N-1
    b=[X(k); vx(k)];
    LU= linsolve(A,b);
    X(k+1) = LU(1);
    VX(k+1) = LU(2);
end

plot(t,x)
plot(t,VX)
plot(X,VX)
plot(t,E)
```

Note

$E = E_{potencial} + E_{cinética}$

$E = \frac{1}{2} k x^2 + \frac{1}{2} m v^2$

Este sistema pode ser programado?

1) Use um **ciclo For**, por exemplo.

2) Use o **LINSOLVE** do Matlab

Neste caso é necessário reescrever o sistema, isolando no membro esquerdo os termos c/ índice $k+1$ e no membro direito os termos c/ índice k,

$$\begin{cases} x_{k+1} + \left(-\frac{h}{2}\right) vx_{k+1} = x_k + \frac{h}{2}\, vx_k \\ \omega^2 \frac{h}{2} x_{k+1} + vx_{k+1} = vx_k - \frac{h}{2}\omega^2 x_k \end{cases}$$

na forma matricial, tem-se

$$\underbrace{\begin{bmatrix} 1 & -\frac{h}{2} \\ \omega^2 \frac{h}{2} & 1 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_{k+1} \\ vx_{k+1} \end{bmatrix}}_{X} = \underbrace{\begin{bmatrix} x_k + \frac{h}{2} vx_k \\ -\omega^2 \frac{h}{2} x_k + vx_k \end{bmatrix}}_{b}$$

A é matriz constante.

E Matlab tem-se:

Ete > definir constantes
definir vetores
como fez na série 1

```
A = [ 1  -h/2 ; w^2*h/2  1 ];

for k = 1 . . .
    b = [x(k) + vx(k)*h/2 ; -w^2*h/2*x(k) + vx(k)]
    LU = linsolve(A,b);
    x(k+1) = LU(1);
    vx(k+1) = LU(2);
end
```

plot's _ _ _ _