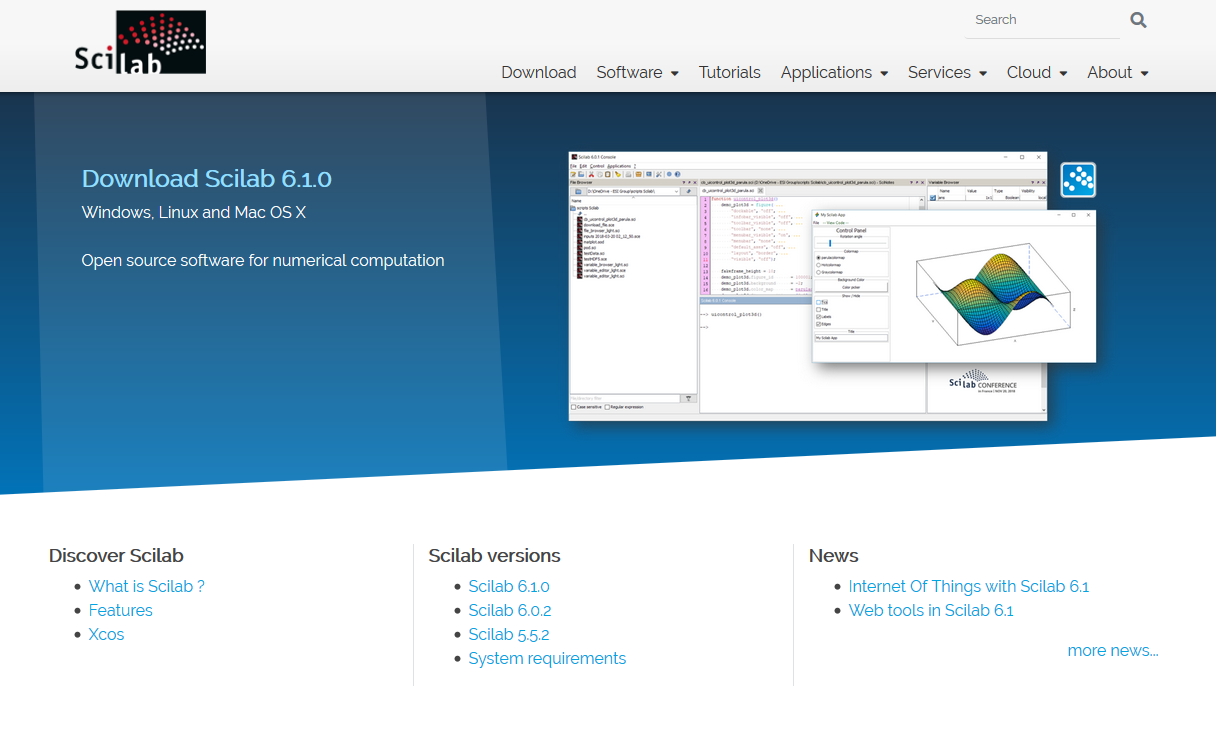
# Introdução

O Scilab e um software para computação cientifica, gratuito, com código fonte aberto e interface para a linguagens **FORTRAN** e **C.** Elepermite a solução de problemas numéricos com uma fração do tempo que seria necessário escrever um programa em uma linguagem como **FORTRAN**, **PASCAL** ou **C**, devido as sua centenas funções matemáticas.

Como instalar esses programas ?

O scilab é encontra varias versão na internet mas sugiro instalar sempre o mais recente (<https://www.scilab.org/>).



## Folha 1 – Introdução ao Scilab e python

### Preparaçao teórica

Considere a seguinte matriz 2x2 :



1. Calcule B = A \* A;
2. Calcule C = produto ponto a pont de A com A ;
3. Calcule A-1;
4. Calcule o inverso ponto a ponto de A , 1/A ;
5. Faça uma Rotação vertical das linhas da matriz;
6. Faça uma rotação horizontal das colunas da matriz .

Para scilab :

A = [2 1 ; 1 2 ];

B = A \* A ;

C = A .\* A;

D = inv(A);

F = 1/A;

E = A(:,$:-1:1);

F = A($:-1:1,:);

### Trabalho prático

Exercicios 1 : Declaração de matrizes e funções.

1.declare um eixo temporal discreto t de dimensão N = 1024 para t1 = 0, …,tN = 2 segundos, utilizando a função *linspace* (faça um estudo rápido dela). Qual o intervalo de tempo entre cada amostra em segundos ? Procure uma outra forma de fazer o mesmo eixo temporal sem utilizar a função *linspace.* E se quiser fazer um eixo temporal de [-2;2 ] segundos?

Para scilab :

N = 1024; *//dimensao da amostra*

ti = 0; *// tempo inicial*

tf = 2 ; *// tempo final*

intervalo\_tempo = linspace(ti,tf,N);

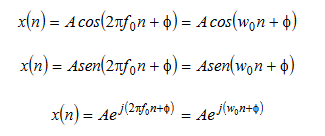
distancia\_amostras = intervalo\_tempo(2) - intervalo\_tempo(1);

*// fazer o mesmo processo mas sem usar linspace*

intervalo\_tempo\_v2 = [ti: 4/N: tf];

distancia\_de\_amostras\_v2 = intervalo\_tempo\_v2(2) - intervalo\_tempo\_v2(1);

2.Construa agora uma função s(*t*) = sin(x) onde x = ω0 *t* . Se quisermos um seno de frequência *f*0 = 10 Hz, como fazer? Construa o vector s em função do vector *t* do numero anterior.



Para scilab:

Cria um ficheiro com nome “st.sce " com o seguinte código :

function [**x**]=st(**frequencia**, **tempo**)

**x** = (2\*%pi\***frequencia**) .\* **tempo**

endfunction

Depois voltamos ao ficheiro que crias-te o intervalo de tempo :

frequencia = 10;

st\_1 = st(frequencia,intervalo\_tempo);

st\_2 = st(frequencia,intervalo\_tempo\_v2);

3. Queremos aumentar a amplitude da função s(t) para um valor A = 4.3.

Para scilab:

Amplitude = 4.3;

st\_1 = Amplitude .\* st\_1;

st\_2 = Amplitude .\* st\_2;

4. vamos agora construir um nova função g(t) = exp(- α *t*), onde α = 2 . Determine o vector g .

Para scilab:

Cria um ficheiro com nome “gt.sce " com o seguinte código :

*// gt(alpha,tempo)*

function [**x**]=gt(**alpha**, **tempo**)

lomba = -**alpha** .\* **tempo**

**x** = exp(lomba)

endfunction

Depois voltamos ao ficheiro que crias-te o intervalo de tempo :

alpha = 2 ;

gt\_1 = gt(alpha,intervalo\_tempo);

gt\_2 = gt(alpha,intervalo\_tempo\_v2);

5. utilizando a função plot represente os vetores s e g , primeiro em duas figuras separadas, depois no mesmo gráfico e a função subplot. Coloque legendas nos eixos do tempo e amplitude e títulos no vários gráfico:

Para scilab :

subplot(2,2,1)

title('Sinal Continuo senoidal','fontsize',5,'color','blue')

xlabel('Tempo','color','blue')

ylabel('Amplitude(HZ)','color','blue')

plot(intervalo\_tempo, st\_1);

subplot(2,2,2)

title('Energia','fontsize',5,'color','red')

xlabel('Tempo','color','red')

ylabel('Amplitude(HZ)','color','red')

plot(intervalo\_tempo,gt\_1,'color','red');

subplot(2,2,3)

title('Sinal Continuo senoidal','fontsize',5,'color','blue')

xlabel('Tempo','color','blue')

ylabel('Amplitude(HZ)','color','blue')

plot(intervalo\_tempo\_v2, st\_2);

subplot(2,2,4)

title('Energia','fontsize',5,'color','red')

xlabel('Tempo','color','red')

ylabel('Amplitude(HZ)','color','red')

plot(intervalo\_tempo\_v2,gt\_2,'red')

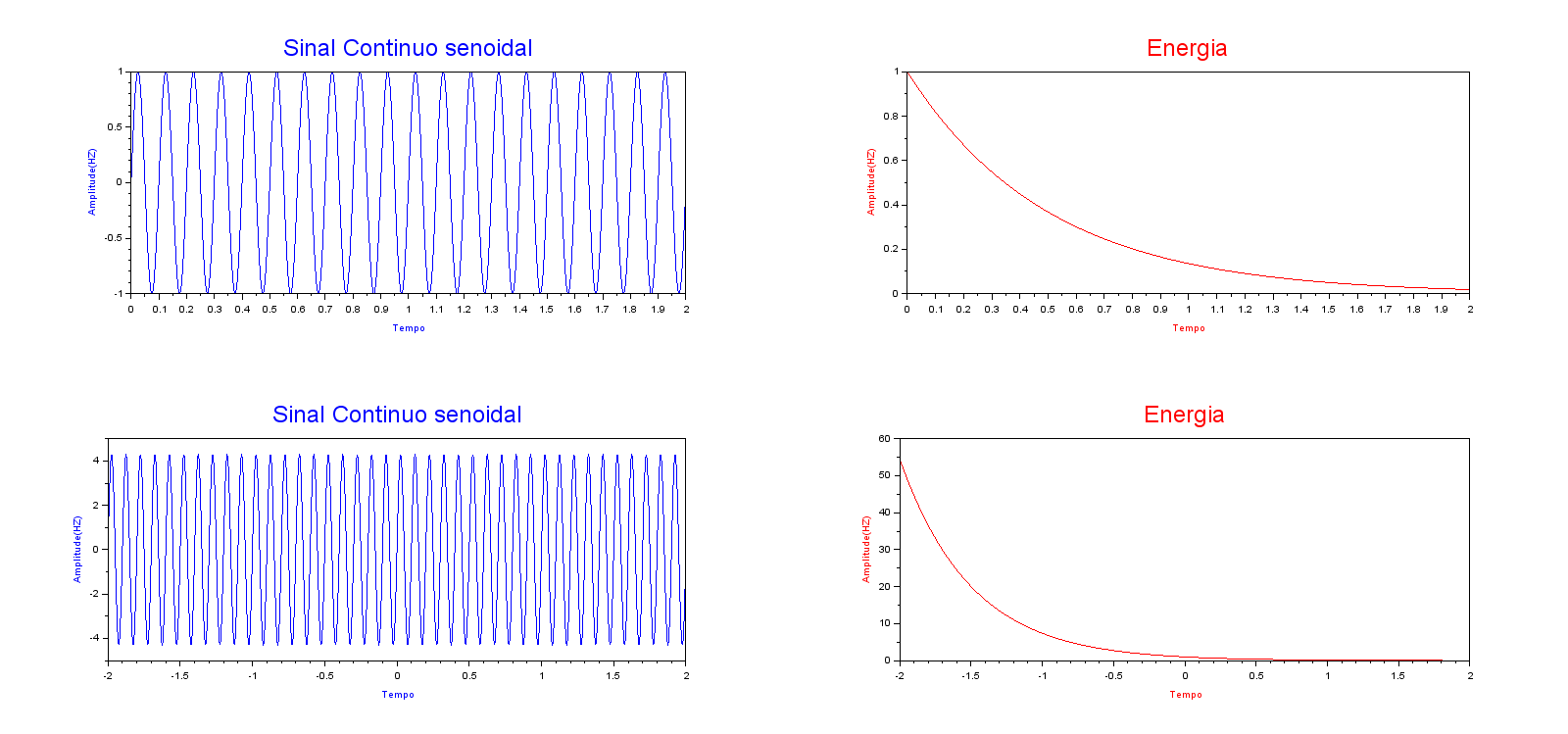


Figura 1. O grafico do codigo acima

6. Determine agora a função h(t) = g(t)s(t). Represente h(t).

ht\_1 = gt\_1 \* st\_1;

ht\_2 = gt\_2 \* st\_2;

scf(g2);

subplot(2,1,1)

title('Resposta final','fontsize',5,'color','green')

xlabel('Tempo','color','green')

ylabel('Amplitude(HZ)','color','green')

plot(intervalo\_tempo,ht\_1,'color','green');

subplot(2,1,2)

title('Resposta final','fontsize',5,'color','green')

xlabel('Tempo','color','green')

ylabel('Amplitude(HZ)','color','green')

plot(intervalo\_tempo\_v2,ht\_2,'color','green');

Exercício 2 : Funções particulares

*// n0 localizaçao*

*// n1 tempo inicial*

*// n2 tempo final*

function [**x**, **n**]=dirac\_unit(**n1**, **n2**, **n0**)

amo = **n2** - **n1** +1;

**n**= linspace(**n1**,**n2**,amo);

**x** = **n** - **n0** >= 0 ;

endfunction

## Follha 2 – Sinais discretos em Scilab

### Preparaçao teórica

Exercicio 1 : considere o sinal x [n]