

O Poderoso SciLab - Parte III

Mário Leite

...

Como já havia comentado nas partes I e II desta trilogia, o SciLab é uma poderosa ferramenta de computação numérica, baseada no MATLAB, mas com a vantagem de ser *open source* e *free*. Assim sendo, pode ser baixada e utilizada livremente; bastando, apenas, obter uma licença de uso da **CeCILL (CeA CNRS INRIA Logiciel Libre)** sem nenhum custo adicional.

Como foi mostrado nas duas partes anteriores, esta ferramenta pode resolver os mais variados tipos de problemas: desde os cálculos mais simples até os mais complexos, passando pela criação de gráficos, resolução de equações ordinárias e diferenciais, programação baseada numa linguagem própria, que é uma mistura de sintaxe das linguagens **Basic** (aceita *then/else/end*) e **C** (aceita *printf*).

Quanto à criação e plotagem de gráficos, o SciLab é um caso à parte: baseado em funções simples de usar, ele pode criar gráficos e desenhos fantásticos; dependendo apenas da vontade e habilidade do programador. O programa **“PasseioCopacabana”** simula as ondas desenhadas na famosa calçada de Copacabana (Rio) usando apenas uma função em Pi com *seno* e *coseno*. A **figura 1a** mostra a saída na tela gráfica, e a **figura 1b** a saída no terminal das instruções de *display* e *print*.

Mas, embora o SciLab seja uma ferramenta bem fácil de usar, o usuário deve ter em mente que ao contrário das linguagens “normais” de programação, esta ferramenta sempre apresenta novidades a cada versão e algumas funções podem não ter retrocompatibilidade com as de versões anteriores, e mesmo não existirem mais. Por exemplo, a função **eval()** que converte números em *strings*, e muito usada nas versões 5.xx, não pode ser usada na versão atual **2025.1.0**; então **eval(num)** deve ser alterada para **string(num)**. E isto não ocorre só com funções básicas: pode mudar radicalmente em outras situações; por exemplo, na primeira versão 5.00, existia a função **linpro()** que fazia a minimização de uma função objetivo com muita facilidade, mas nas versões subsequentes ela simplesmente sumiu! Portanto, ao fazer *download* o usuário deve ler com bastante atenção as novidades da versão baixada e verificar se alguma função de versão anterior ainda existe. Mas, mesmo com esses cuidados, o SciLab continua sendo uma das melhores ferramentas de computação numérica, e deve ser fortemente considerada quando for preciso uma ferramenta desse tipo.

O livro **“SciLab: Uma Abordagem Prática e Didática”**, do autor, pode ser uma grande ajuda para os programadores iniciantes que querem aprender sobre esta ferramenta.

Veja mais detalhes deste livro no *link*:

<https://www.amazon.com.br/SciLab-Uma-Abordagem-Pr%C3%A1tica-Did%C3%A1tica/dp/8539906570>

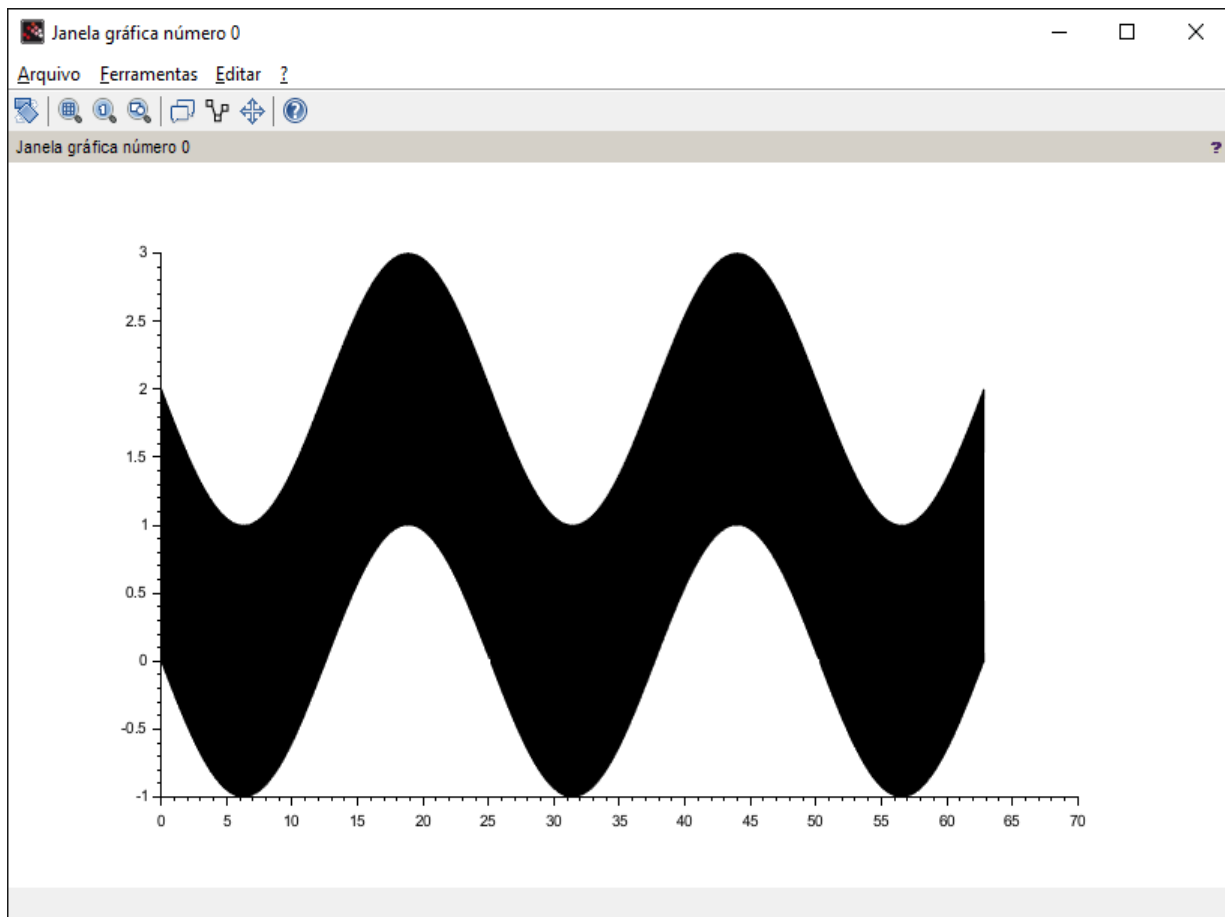


Figura 1a - Saída do programa em tela gráfica

```
Scilab 2025.1.0 Console
Arquivo  Editar  Controle  Aplicativos  ?
[Icons]
Scilab 2025.1.0 Console
--> exec('D:\Cantinho da Programação\Códigos\SciLab\PasseioCopacabana.sce', -1)
    "Processando... Aguarde!"

    "Terminou..."
--> exec('D:\Cantinho da Programação\Códigos\SciLab\PasseioCopacabana.sce', -1)
    "Processando... Aguarde!"

Terminou...
--> |
```

Figura 1b - Saída do programa no terminal

```

//PasseioCopacabana.sce
//Simula o desenho das ondas na calçada de Copacabana (Rio)
//-----

clc; //limpa a tela do monitor de vídeo
x = (0:0.01:20*pi); //define o período da função para as ondas
cor = 1; //define a cor das ondas na calçada (preto no fundo branco)
clf(); // limpa a janela gráfica

disp("Processando... Aguarde!")

printf("\n");
f = gcf();
if (cor==1) then
    f.background = color("white");
    corLinha = color("black");
    for j=1:200
        y = cos(j*x/4) + (1-sin(x/4));
        plot2d(x, y, corLinha)
    end
else
    break //aborta o processamento (cor inválida)
end

printf("\n")
printf("Terminou...")
//Fim do programa "PasseioCopacabana.sce"

```