**Capítulo 1 - Introdução ao Scilab**

**1.1 - O Ambiente Scilab**

 A unidade de dados fundamental em um programa Scilab é o array. Um array é uma coleção de valores de dados organizados em linhas e colunas, sendo conhecidos por um único nome. Os valores de dados individuais dentro de um array podem ser acessados incluindo o nome do array, seguido por subscritos entre parêntesis, que identificam a linha e a coluna do valor particular. Mesmo os valores escalares são tratados pelo Scilab como arrays; eles são simplesmente arrays com uma linha e uma coluna.

 Quando o Scilab está em execução, ele pode exibir vários tipos de janelas que aceitam comandos ou exibem informações. As três tipos mais importantes janelas são a Janela do Console, onde se faz a entrada de comandos; a Janela de Figuras, a qual exibe plotagens e gráficos; e a Janela de Edição (SciNotes), que permite ao usuário criar e modificar programas Scilab.

 Além destas, o Scilab pode exibir outras janelas que proveem ajuda e que permitem ao usuário examinar os valores das variáveis definidas na memória.

 Quando você inicia o Scilab 5.4.1, você tem a visualização do ambiente de trabalho do Scilab. Este ambiente de trabalho contém as janelas que exibem os dados Scilab, além de menus de opções e barras de ferramentas. A maioria das janelas estão fixadas, porém o usuário pode separá-las em janelas individuais conforme suas necessidades (vide a ajuda do Scilab).

 A configuração default do ambiente de trabalho do Scilab é exibida na Figura 1.1. Ela integra várias ferramentas para gerenciamento de arquivos, variáveis e aplicações dentro do ambiente Scilab.

 As principais janelas acessíveis a partir do ambiente de trabalho do Scilab são:

* A Janela do Console
* O Histórico de Comandos
* A Janela de Edição - SciNotes
* O Navegador de Arquivos
* O Navegador de Variáveis

**1.2 - A Janela do Console**

 A janela central do ambiente de trabalho é a Janela do Console. Ela permite que o usuário insira comandos interativamente com o Scilab através do prompt ( **--->** ), os quais serão executados ao se pressionar a tecla **<ENTER>**.



Figura 1.1 – O ambiente de trabalho default do Scilab. A exata aparência do ambiente de trabalho pode diferir de um tipo de computador para outro, dependendo de suas configurações.

 Como um exemplo simples de cálculo interativo, suponha que você deseje calcular a área de um círculo, com raio de 2,5 m. Isto pode ser feito na Janela de Comandos Scilab digitando-se:

**---> area = %pi \* 2.5^2**

**area =**

 **19.6350**

 O Scilab calcula a resposta assim que a tecla **<ENTER>** for pressionada e armazena a resposta em uma variável (de fato, um array 1x1) chamada **area**. O conteúdo da variável é exibido na Janela o Console ao término do cálculo. Note que o valor de π é predefinido em Scilab, desta forma, podemos simplesmente usar **%pi** sem a necessidade de declará-lo como sendo 3,141592...).

 Se uma instrução é demasiadamente longa para ser digitada em uma única linha, esta instrução pode ser continuada em linhas sucessivas através da digitação dos três pontos (**...**) ao final da primeira linha, e continuando a seguir nas próximas linhas. Por exemplo, na Figura 1.2, as instruções que calculam x1 e x2 são idênticas:



Figura 1.2 – Instruções para o cálculo de x1 e x2

**1.3 - A Janela de Histórico de Comandos**

 A Janela de Histórico de Comandos exibe uma lista dos comandos que o usuário executou na Janela do Console. Os comandos ficam na lista até serem deletados. Para executar novamente um comando, basta efetuar um clique duplo com o botão esquerdo do mouse. Para deletar um ou mais comandos da Janela de Histórico de Comandos, selecione o comando e efetue um clique com o botão direito do mouse. Um menu *popup* será exibido e permitirá a exclusão do comando, veja a figura 1.3.



Figura 1.3 – A Janela de Histórico de Comandos, mostrando a deleção de um comando

**1.4 - A Janela de Edição - SciNotes**

 A Janela de Edição é usada para criação de novos arquivos, programas Scilab, ou para modificação de arquivos existentes. Os seguintes passos são realizados para criação de um arquivo no SciNotes:

1. Clique no ícone referente ao SciNotes:



1. Digite o programa na Janela do SciNotes;
2. Clique no ícone para salvar o arquivo; forneça um nome de arquivo com a extensão **sce**



1. Escolha o diretório para salvar o arquivo:



1. Clique no ícone para executar o programa e veja o resultado exibido na Janela do Console:



Resultado na Janela Console:



**1.5 - O Ambiente de Trabalho do Scilab**

 Um comando **x = 20** cria uma variável denominada por **x**, armazena nela o valor **20**, e a salva em uma parte da memória do computador conhecida como Ambiente de Trabalho. O Ambiente de Trabalho é uma coleção de todas as variáveis, e arrays, que podem ser utilizados em um comando particular ou em um programa Scilab. Todos os comandos, e arquivos, executados na Janela do Console, compartilham um ambiente de trabalho comum, logo eles compartilham todas as variáveis. A Janela do Navegador de Variáveis exibe todas as variáveis do ambiente em um dado momento.

 Uma lista de variáveis e arrays armazenados no ambiente de trabalho corrente pode ser gerada com o comando **whos**. A lista de variáveis é exibida na Janlea do Console, bem como no Navegador de Variáveis. Exemplificando, seja o seguinte programa Scilab:

**raio = 2; volume = (4/3) \* %pi \* raio^3;**

**-->whos**

**Nome Tipo Tamanho Bytes**

**• • •**

**%T boolean 1 por 1 24**

**%t boolean 1 por 1 24**

**• • •**

**raio constant 1 por 1 24**

**• • •**

**volume constant 1 por 1 24**

**whos function 15416**

**• • •**



 Note que as variáveis raio e volume estão no mesmo ambiente de trabalho, podendo ser usadas por qualquer programa Scilab.

 O conteúdo de qualquer variável do ambiente de trabalho pode ser determinado digitando-se o nome da variável na Janela do Console, veja a Figura 1.4.



Figura 1.4 – Exibindo o conteúdo de uma variável

 Também podemos visualizar o conteúdo de uma variável armazenada na memória do computador, ou seja, no ambiente de trabalho, através do comando **disp**, veja a Figura 1.5.



Figura 1.5 – Exemplo de uso do comando **disp**

 Uma variável pode ser deletada, ou apagada, do ambiente de trabalho através do comando **clear**:

**clear var1, var2, ...**

onde **var1** e **var2** são nomes de variáveis a serem deletadas. O comando **clear**, sem mencionar as variáveis, limpa todas as variáveis do ambiente de trabalho, veja as Figuras 1.6 e 1.7.



Figura 1.6 – Exemplo de uso do comando **clear**



Figura 1.7 – Exemplo de uso do comando **clear**

**1.6 - Buscando Ajuda**

 A forma mais simples de buscar ajuda no Scilab é através do Navegador de Ajuda, Figura 1.8 e 1.9:



Figura 1.8 – Ícone do Navegador de Ajuda

 Através do Navegador de Ajuda pode-se consultar os detalhes de funcionamento de um comando particular. Por exemplo, consultando-se o comando **clc**:



Figura 1.9 – Navegador de Ajuda

 Também, pode-se digitar no Console o comando exibido abaixo, obtendo-se a mesma janela da Figura 1.9.



 Caso o comando **help** seja digitado sem argumento, ou seja, sem nenhum nome de comando após a palavra help, será exibida a mesma janela da Figura 1.9.

**1.7 - Alguns Comandos Importantes**

**1.7.1 - clc**

 O comando **clc** limpa os conteúdos, impressões, da Janela do Console do Scilab.

**1.7.2 - clear**

 O comando **clear** limpa o Ambiente de Trabalho do Scilab, ou seja, as variáveis e seus respectivos valores.

**1.3.7.2 - abort**

 No *prompt* do Console, digitamos os comandos a serem processados pelo Scilab. Em situações normais, o *prompt* ( **-->** ) é exibido da seguinte forma:



 Entretanto, pode ocorrer erros que alteram a exibição do *prompt*:



 Nesta situação, o usuário pode digitar o comando **abort** para retornar ao estado inicial do *prompt*. O comando **resume** retoma a execução da tarefa que estava sendo executada, mas não interrompe a ocorrência do erro.



**1.8 - Usando o Scilab Como Um Bloco de Notas**

 Em sua forma mais simples, o Scilab pode ser usado com um bloco de notas para efetuar cálculos. Os cálculos são realizados digitando-se diretamente no *prompt* as expressões matemáticas. A seguir, algumas operações matemáticas e suas respectivas representações simbólicas no Scilab:

|  |  |  |
| --- | --- | --- |
| **Operação Matemática** | **Representação no Scilab** | **Exemplo** |
| Adição | **+** | **2 + 8** |
| Subtração | **-** | **3 - 9** |
| Multiplicação | **\*** | **19 \* 7.8** |
| Divisão | **/** | **8.88 / 0.0001** |
| Potenciação | **^** | **2^(1/3)** |

 Assim que uma expressão matemática, ou comando do Scilab, for digitado no *prompt*, e a seguir, for teclado **<ENTER>**, o resultado será exibido no Console.

 Por exemplo, seja o cálculo a área de uma circunferência dada pela fórmula:

$$A= π r^{2}$$

onde r é o raio da circunferência. Supondo que o raio seja 5 cm, temos:



**Observações:**

* tudo que for escrito após **//** é ignorado pelo Scilab. Esta é a forma de se escrever um comentário em um código Scilab. O comentário tem a função de esclarecer alguma coisa importante para o programador.
* na linha que temos o comentário 1, ocorre a definição de uma variável **r**, a qual recebe um valor, no caso **5**. A partir deste ponto, a variável **r** passa a existir no ambiente do Scilab, representando uma área da memória do computador, a qual é acessada pelo nome da variável, **r**. Após teclado **<ENTER>** ocorre a exibição do conteúdo armazenado em **r**, ou seja, **r = 5**.
* na linha do comentário 2, definimos uma variável para a área da circunferência, **A**, a qual receberá o valor resultante da expressão matemática **%pi \* r^2**. No final da linha colocou-se um ponto e vírgula, **;** , o que impediu a exibição do resultado da atribuição do valor da expressão a variável **A**.
* **%pi** é uma valor pré-definido em Scilab referente ao valor de **π**.
* na linha do comentário 3, utilizou-se o comando **disp**, para exibir o conteúdo armazenado na posição de memória designada pela variável **A**.

 Caso desejemos calcular o volume de um cilindro cuja base é uma circunferência de área igual a área armazenada em A, definimos a altura do cilindro, h = 4, e a equação para o volume:



 Agora, **volume** representa uma variável na memória, a qual armazena o valor **314.15927** (o resultado do cálculo da expressão matemática).

**1.9 - Exercícios**

**Exercício 1**

 A distância percorrida por uma bola em queda livre no ar é dada pela equação:

$$x= x\_{0}+ v\_{0} t+ \frac{1}{2} a t^{2}$$

 Utilize o Scilab para calcular a posição da bola no tempo t = 5 s, se x0 = 10 m, v0 = 15 m/s e a = -9,81 m/s2 .

**Exercício 2**

 Suponha que x = 3 e y = 4. Utilize o Scilab para avaliar as seguintes expressões matemática:

$$\frac{x^{2} y^{3}}{(x-y)^{2}}$$

$$\frac{1}{x^{2}- y}- e^{-4x}+\sqrt[3]{35/y}\* \sqrt{x\*y}$$

$$\sin(\left(2x\right)\*5y)+ (\cos(3.8765+x))^{0.5}- \frac{ln⁡(x+y)}{log⁡(x-y)} $$

**1.10 - Mais Comandos do Scilab**

**1.10.1 - Formatação Para Exibição de Números**

 No Scilab tem-se dois tipos de formatação para os números números: Formato Variável e Formato Científico.

**Formato Variável**

 É a formatação padrão, com tamanho máximo de 10 posições para o número exibido, reservando uma posição para o ponto decimal e outra para o sinal. Por exemplo:

**-->x = 1.3456789012345**

 **x =**

 **1.3456789**

o numero impresso tem 10 posições, sendo uma para o sinal:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| + | 1 | . | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

 Além disso, é possível definir a saída de um processamento numérico em função de seu tamanho, através do comando **format**:

**format(n)**

onde **n** é o tamanho total, incluindo o ponto decimal e o sinal. Por exemplo:

**-->format(15)**

**-->x**

 **x =**

 **1.345678901235**

redefine o formato para o tamanho 15 (com doze decimais)

**Formato Científico**

 O comando

**format('e')**

redefine o formato para **'e'**, com a saída exibida no formato científico. O valor exibido é truncado na oitava casa decimal, onde D+00 significa “10 elevado a 0” que é igual a 1. Por exemplo:

**-->format('e')**

**-->x**

 **x =**

 **1.34567890D+00**

 Agora, vamos redefinir a saída padrão com 10 posições:

**-->format('v', 10)**

 **-->x**

 **x =**

 **1.3456789**