

Pengenalan Scilab - *Freeware* Untuk Komputasi Numerik

Saifuddin Arief
Saifuddin.Arief@rocketmail.com

Lisensi Dokumen:

Copyright © 2003-2018 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Scilab adalah *freeware* yang dikembangkan untuk komputasi numerik. Scilab menyediakan ratusan fungsi yang merepresentasikan operasi matematika, analisis data serta algoritma dalam komputasi numerik. Scilab juga merupakan bahasa pemrograman tingkat tinggi yang dapat digunakan untuk pengembangan suatu algoritma. Scilab tersedia untuk sistem operasi Windows, Linux dan MacOS. Alamat situs Scilab adalah <http://www.scilab.org>. Situs tersebut menyediakan dokumentasi dan informasi tentang Scilab serta file instalasinya. Scilab yang digunakan dalam penulisan artikel ini adalah Scilab versi 6.0.1.

Dasar-Dasar Penggunaan Scilab

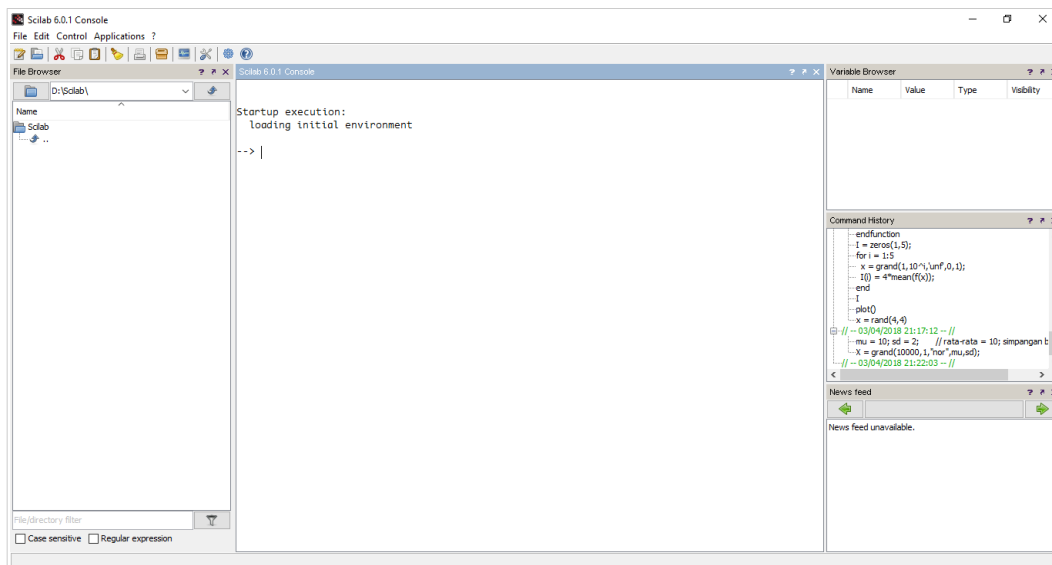
Scilab dapat dijalankan dari menu Start atau *shortcut* Scilab yang terdapat pada jendela pada jendela Dekstop. Setelah dijalankan perintah tersebut maka akan muncul suatu jendela Scilab, seperti yang ditunjukkan pada Gambar 1.

Simbol `-->` yang terdapat pada jendela Scilab merupakan tanda bahwa Scilab siap untuk menerima suatu perintah. Sebagai contoh, untuk melakukan perhitungan $12 + 34$ maka ekspresi matematika tersebut diketikkan setelah simbol `-->` kemudian menekan tombol `enter` untuk menjalankan perintahnya. Scilab akan menampilkan hasil perhitungannya pada baris berikutnya.

```
--> 12 + 34  
ans =
```

```
46.
```

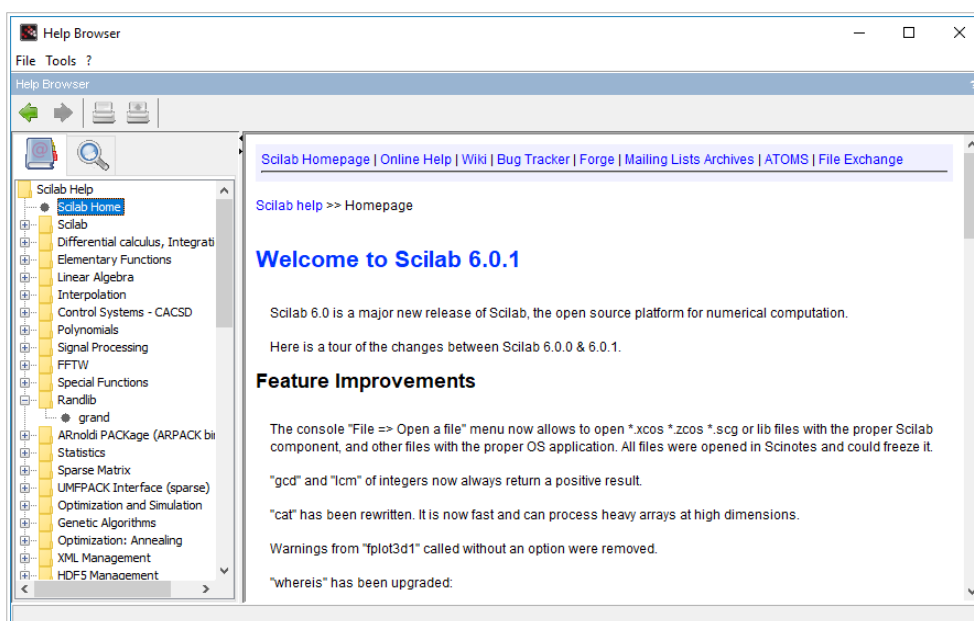
Simbol `-->` yang muncul kembali menunjukkan bahwa Scilab siap untuk mengerjakan suatu perintah berikutnya.



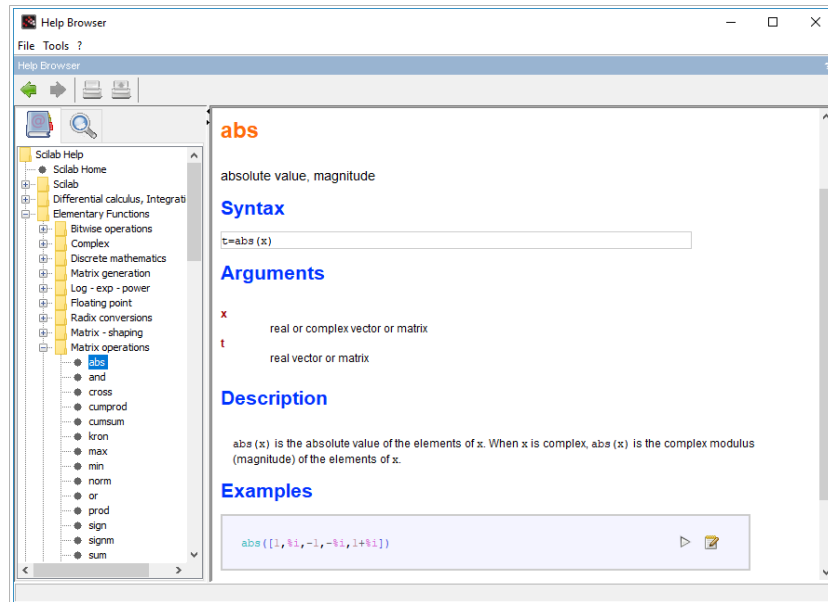
Gambar 1. Scilab

Scilab dilengkapi dengan sistem bantuan yang baik. Jendela sistem bantuan Scilab dapat ditampilkan melalui menu ? - Scilab Help, menekan tombol F1 atau dengan menggunakan perintah help. Pada jendela bantuan, kita dapat memperoleh penjelasan yang detail mengenai suatu fungsi atau operator tertentu.

Penjelasan mengenai suatu perintah, fungsi atau operator tertentu juga dapat juga dilakukan dengan menggunakan perintah help("topik") atau help topik. Sebagai contoh, penjelasan tentang fungsi abs dapat diperoleh dengan perintah help("abs") atau help abs. Apabila dijalankan perintah tersebut maka akan muncul sebuah jendela seperti pada Gambar 3.



Gambar 2. Jendela sistem bantuan Scilab



Gambar 3. Contoh penjelasan detail tentang suatu fungsi

Untuk keluar dari Scilab dapat dilakukan dengan menggunakan perintah `exit`, melalui menu `File - Quit` atau dengan menekan tombol `×` yang terdapat pada ujung kanan atas dari jendela Scilab.

Operator, Fungsi Matematika dan Variabel

Operator-operator untuk perhitungan aritmatika yaitu `+`, `-`, `*`, `/` dan `^`. Simbol tersebut masing-masing melambangkan operasi penjumlahan, pengurangan, perkalian, pembagian dan pemangkatan. Di dalam Scilab juga telah terpasang fungsi-fungsi matematika, seperti `sqrt`, `abs`, `exp`, `sin`, `cos`, `tan` dan lain sebagainya.

```
--> 1010*(sqrt(9.81^2 + 7^2)*((0.07 + 0.0214)*cos(35.5/180*pi)))
ans =
    905.71361
```

Nilai suatu obyek atau hasil perhitungan dapat disimpan ke dalam suatu variabel dengan statemen sebagai berikut:

```
x = ekspresi
```

dimana `x` adalah nama variabel. Dalam penggunaan variabel, huruf kecil dan huruf besar adalah berbeda. Nilai dari suatu ekspresi akan ditampilkan pada baris berikutnya kecuali jika ditambahkan tanda titik koma (`;`) pada akhir ekspresinya.

Di dalam Scilab terdapat beberapa variabel khusus yang menyatakan suatu konstanta matematika, seperti `%pi` untuk $\pi = 3.1415927$, `%i` untuk $i = \sqrt{-1}$ serta `%e` untuk $e = 2.7182818$.

Untuk memperjelas perintah-perintah yang dibuat, dapat ditambahkan suatu baris komentar. Baris komentar dapat ditulis sebagai suatu baris tersendiri atau ditulis di belakang suatu statemen. Kumpulan karakter yang terletak setelah tanda `//` atau diantara tanda `/*` dan `*/` akan dianggap sebagai baris komentar.

```
--> luas = 12.5*8
luas =
    100.
--> d = 53;          // jarak (m)
--> t = 15;         /* waktu tempuh (detik) */
--> // kecepatan rata-rata (m/detik)
--> v = d/t
v =
    3.5333333
```

Bilangan Kompleks

Scilab juga dapat menangani bilangan kompleks dan operasi-operasi matematikanya. Bilangan kompleks dinyatakan dengan notasi $z = x + y*i$ atau $z = x + %i*y$ dimana x adalah komponen real dan y adalah komponen imajineranya.

```
--> z1 = 6 - 8*i
z1 =
    6. - 8.i
--> z2 = 3 + %i
z2 =
    3. + i
--> p = z1 + z2
p =
    9. - 7.i
--> q = z1 - z2
q =
    3. - 9.i
--> z1 * z2
ans =
    26. - 18.i
--> z1/z2
ans =
    1. - 3.i
```

Matrik dan Vektor

Pada dasarnya semua data numerik di dalam Scilab dianggap sebagai suatu matrik. Vektor dan skalar merupakan bentuk khusus dari suatu matrik. Vektor adalah suatu matrik yang hanya mempunyai satu baris atau satu kolom saja, sementara itu skalar adalah suatu matrik yang hanya terdiri dari satu elemen saja.

Pembuatan data matrik dan vektor secara manual dilakukan dengan menggunakan operator kurung siku ([]). Untuk memisahkan elemen yang satu dengan elemen yang lainnya yang terletak pada satu baris dapat digunakan tanda koma (,) atau tanda spasi. Kemudian antara baris yang satu dengan yang lainnya dipisahkan dengan tanda titik koma (;) atau tombol Enter.

Contoh-contoh pembuatan matrik dan vektor adalah sebagai berikut:

```
--> X = [1 2 3; 4 5 6; 7 8 9]
X =

    1.    2.    3.
    4.    5.    6.
    7.    8.    9.

--> v = [1 2 3 4 5];           // vektor baris
--> w = [3;5;2]                // vektor kolom
w =

    3.
    5.
    2.
```

Suatu vektor baris dimana nilai elemen-elemennya berubah secara konstan dari suatu nilai awal sampai nilai akhir tertentu dapat dibuat dengan sintak $i:j:k$. Apabila nilai j sama satu maka notasi tersebut dapat ditulis dengan notasi yang lebih singkat yaitu $i:k$.

```
--> i = 1:10
i =

    1.    2.    3.    4.    5.    6.    7.    8.    9.   10.

--> n = 0:0.25:1
n =

    0.    0.25    0.5    0.75    1.

--> m = 1:2:10
m =

    1.    3.    5.    7.    9.
```

Scilab juga menyediakan sejumlah fungsi yang dapat digunakan untuk membuat matrik-matrik khusus, seperti yang diilustrasikan pada contoh-contoh di bawah ini.

```
--> A = zeros(3,3)           // matrik nol
A =

    0.    0.    0.
    0.    0.    0.
    0.    0.    0.

--> B = ones(2,5)           // matrik satuan
B =

    1.    1.    1.    1.    1.
    1.    1.    1.    1.    1.
```

```
--> Y = eye(3,3)           // Matrik identitas
Y =

    1.    0.    0.
    0.    1.    0.
    0.    0.    1.

--> D = diag(1:4)         // Matrik diagonal
D =

    1.    0.    0.    0.
    0.    2.    0.    0.
    0.    0.    3.    0.
    0.    0.    0.    4.
```

Operasi Berbasis Vektor

Di dalam Scilab, secara umum operasi-operasi matematika terhadap obyek matrik dan vektor dapat dilakukan dengan mudah tanpa harus menggunakan suatu perulangan, seperti yang diilustrasikan pada contoh-contoh di bawah ini.

```
--> x = 0:%pi/4:%pi;

--> cos(x)
ans =

    1.    0.7071068    6.123D-17    - 0.7071068    - 1.

--> u = [1 2 3];

--> y = exp(u)
y =

    2.7182818    7.3890561    20.085537

--> z = log(y)
z =

    1.    2.    3.
```

Operasi Aljabar Linier, seperti penjumlahan, pengurangan dan perkalian, juga dapat dilakukan dengan sangat mudah tanpa harus menggunakan suatu ekspresi perulangan secara eksplisit. Pada operasi aljabar argumen-argumennya harus mempunyai dimensi yang kompatibel, jika dimensinya tidak kompatibel maka operasinya tidak dapat dieksekusi dan akan muncul suatu pesan kesalahan.

```
--> X = [9 8 5; 1 3 0; 0 4 6];

--> Y = [3 2 1; 4 5 6; 9 8 7];

--> A = X + Y
A =

    12.    10.    6.
    5.     8.     6.
    9.    12.    13.
```

```
--> B = X - Y  
B =
```

```
    6.    6.    4.  
   - 3.   - 2.   - 6.  
   - 9.   - 4.   - 1.
```

```
--> Z = A + U
```

Inconsistent row/column dimensions

```
--> p = [1 2; 3 8];
```

```
--> j = [6; 4];
```

```
--> k = ones(3,3);
```

```
--> g = p*j  
g =
```

```
    14.  
    50.
```

```
--> h = p*k
```

Inconsistent row/column dimensions.

Di dalam Scilab dapat juga dilakukan operasi perkalian dan pembagian terhadap obyek matrik dan vektor dengan melakukan operasi secara elemen dengan elemen. Notasi untuk perkalian dan pembagian secara elemen dengan elemen adalah `.*` dan `./`.

```
--> x = [1 2; 3 4];
```

```
--> y = [5 6; 7 8];
```

```
--> u = x.*y  
u =
```

```
    5.    12.  
   21.    32.
```

```
--> z = y./x  
z =
```

```
    5.    3.  
 2.3333333  2.
```

Visualisasi Data

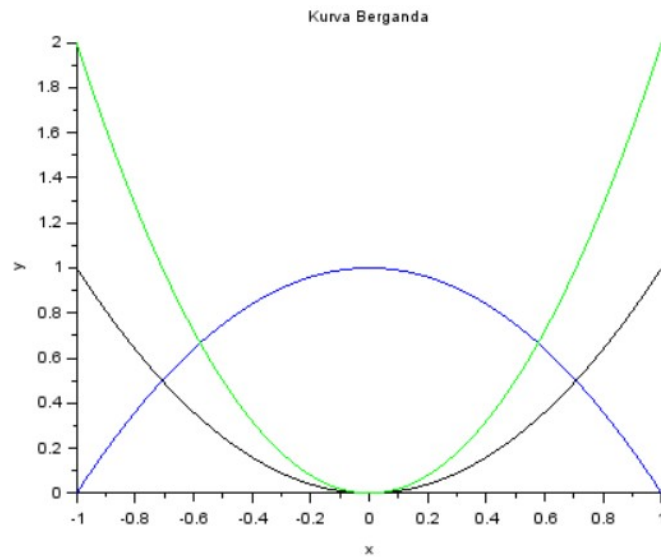
Scilab dapat digunakan untuk melakukan visualisasi data, baik secara dua dimensi maupun tiga dimensi. Untuk membuat grafik dua dimensi kita dapat menggunakan perintah `plot2d`, kemudian untuk grafik tiga dimensi kita dapat menggunakan perintah `plot3d`, seperti yang diperlihatkan pada contoh-contoh di bawah ini.

```
--> x = linspace(-1,1,61)';
```

```
--> y1 = x.^2; y2 = 1 - y1; y3 = 2*y1;
```

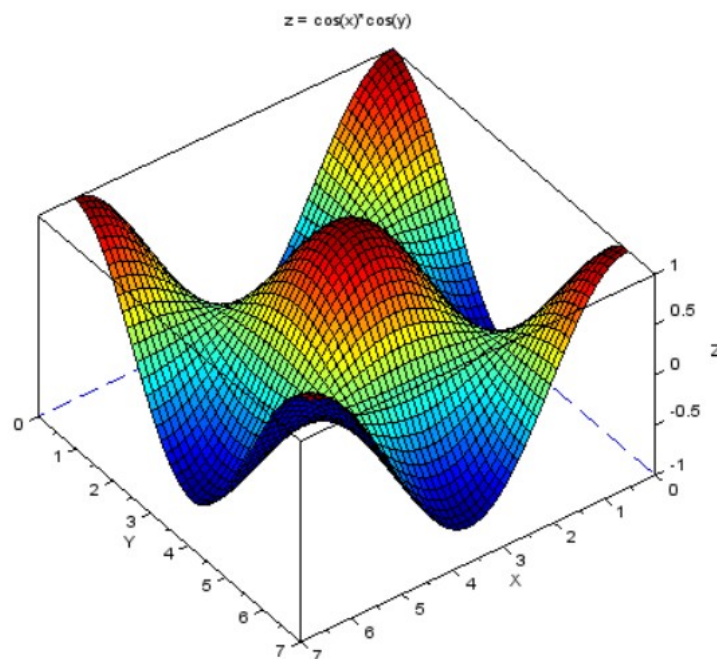
```
--> plot2d(x,[y1 y2 y3])           // Lihat Gambar 4
```

```
--> xtitle('Kurva Berganda', 'x', 'y')
```



Gambar 4

```
--> x = linspace(0, 2*pi, 50); y = x;  
--> z = cos(x')*cos(y);  
--> f = scf(0); clf;  
--> plot3d1(x, y, z); // Lihat Gambar 5  
--> xtitle('z = cos(x)*cos(y)');  
--> f.color_map = jetcolormap(50);
```



Gambar 5

Perulangan dan Kondisional

Scilab menyediakan sejumlah kontrol pemrograman yang dapat digunakan untuk mengatur jalannya eksekusi suatu program dengan menggunakan statemen perulangan dan kondisional. Ilustrasi penggunaan statemen perulangan `for` dan kondisional `if` diberikan pada contoh-contoh di bawah ini.

```
--> // Contoh statemen perulangan

--> H = zeros(4,4);

--> for i=1:4
>   for j=1:4
>     H(i,j) = 1/(i+j-1);
>   end
> end

--> H
H =

    1.0000000    0.5000000    0.3333333    0.2500000
    0.5000000    0.3333333    0.2500000    0.2000000
    0.3333333    0.2500000    0.2000000    0.1666667
    0.2500000    0.2000000    0.1666667    0.1428571

--> // Contoh statemen kondisional

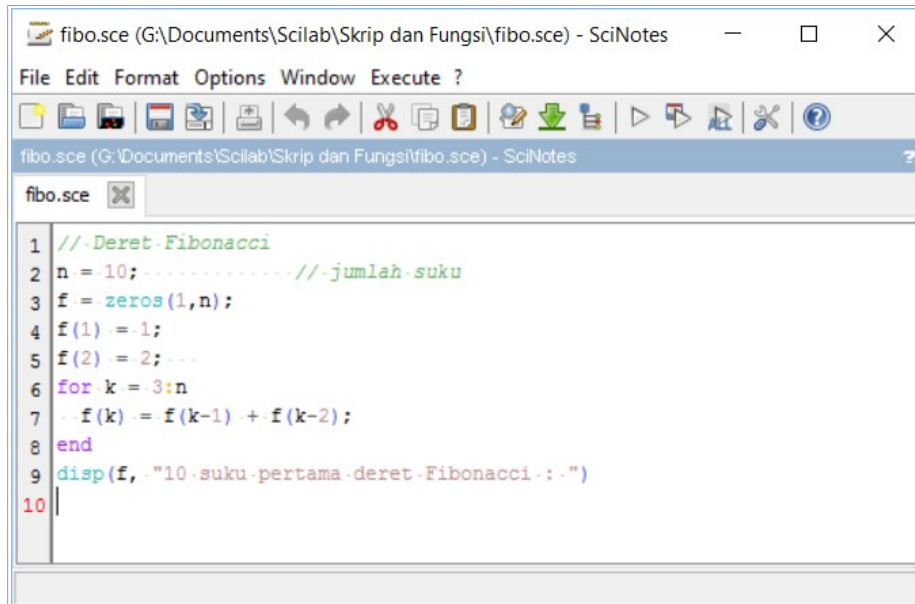
--> function n=nilai(kode)
> // Fungsi untuk mengkonversi nilai dari abjad menjadi angka
>   if kode=='A'
>     n=4;
>   elseif kode=='B'
>     n=3;
>   elseif kode=='C'
>     n=2;
>   elseif kode=='D'
>     n=1;
>   else
>     n=0;
>   end
> endfunction

--> n1 = nilai('A')
n1 =

    4.
```

Skrip

Skrip adalah sebuah file teks yang di dalamnya terdapat perintah-perintah Scilab. Apabila suatu skrip dijalankan maka perintah-perintah yang terdapat di dalamnya akan dieksekusi seperti seolah-olah kita mengetikkannya pada jendela perintah. Skrip dapat dibuat dengan menggunakan teks editor yang telah terpasang pada Scilab, yaitu SciNotes, atau dengan menggunakan program teks editor yang lain seperti Geany, Notepad2 atau ConText. Gambar 6 adalah contoh sebuah skrip (`fibonacci.sce`) yang menggambarkan perhitungan 10 suku pertama dari deret Fibonacci.



Gambar 6

Statemen-statemen yang terdapat di dalam sebuah skrip yang sedang dibuka pada jendela SciNotes dapat dijalankan melalui menu Execute. Sebagai contoh, apabila skrip [fibonacci.sce] dijalankan melalui menu Execute - Save and Execute maka tampilan berikut ini akan muncul pada Jendela Scilab.

```
--> exec('G:\Documents\Scilab\Skrip dan Fungsi\fibonacci.sce', -1)
```

10 suku pertama deret Fibonacci :

1. 2. 3. 5. 8. 13. 21. 34. 55. 89.

Sebuah file skrip yang telah tersimpan pada sebuah direktori juga dapat dieksekusi dengan statemen `exec(fskrip)` dimana `fskrip` adalah nama filenya. Jika filenya tersimpan pada direktori kerja maka argumen `fskrip` cukup dinyatakan dengan nama filenya saja. Namun jika filenya tidak tersimpan pada direktori kerja maka argumen `fskrip` harus berupa nama file skrip dan nama direktorinya.

Fungsi

Fungsi merupakan kumpulan dari statemen-statemen Scilab yang dapat melakukan suatu komputasi atau perhitungan tertentu. Fungsi dapat dibuat dalam bentuk suatu file atau secara inline pada jendela perintah. Bentuk umum dari suatu fungsi adalah sebagai berikut:

```
function [out1,out2,...] = fname(in1,in2,...)
    statemen-statemen
endfunction
```

dimana `fname` adalah nama fungsi, dan `in1, in2, ...` adalah argumen-argumen input serta `out1, out2, ...` adalah argumen-argumen output.

Suatu file-fungsi yang telah selesai dibuat dengan teks editor SciNotes dapat disimpan melalui menu File - Save. Meskipun dapat digunakan sembarang ekstensi namun sebaiknya suatu file fungsi disimpan dengan ekstensi `sci`. Ilustrasi dari sebuah file fungsi diberikan pada gambar 7.

```

1 function [x, fx] = bisect(f, x1, x2, tol)
2   ...Penentuan akar persamaan nonlinier dengan metode Bisection
3
4   ...if f(x1) == 0.0; x = x1; return; end
5   ...if f(x2) == 0.0; x = x2; return; end
6   ...if sign(f(x1))*sign(f(x2)) > 0
7   ... error('Akar tidak terdapat di dalam selang [x1,x2]')
8   ...end
9
10  ...n = ceil(log(abs(x2 - x1)/tol)/log(2.0)); ...Iterasi maksimum
11
12  ...for i = 1:n
13  ... x3 = 0.5*(x1 + x2);
14  ... f3 = f(x3);
15  ... if f(x3) == 0.0
16  ...   x = x3; return
17  ... end
18  ... if sign(f(x2))*sign(f(x3)) < 0.0
19  ...   x1 = x3; f1 = f3;
20  ... else
21  ...   x2 = x3; f2 = f3;
22  ... end
23  ...end
24  ...x = (x1 + x2)/2;
25  ...fx = f(x);
26  ...endfunction
27 endfunction
28

```

Gambar 7. Contoh File Fungsi

Berikut ini contoh pembuatan suatu fungsi secara inline.

```

--> function y = h(x)
>   y = x^3 - x - 3
> endfunction

```

Fungsi-fungsi baru yang dibuat dapat digunakan sebagaimana fungsi-fungsi yang telah terpasang pada Scilab. Untuk fungsi-fungsi yang dibuat dalam suatu file fungsi maka harus dipanggil terlebih dahulu ke dalam ruang kerja dengan menggunakan perintah `exec`, seperti yang ditunjukkan pada contoh di bawah ini.

```

--> exec('G:\Documents\Scilab\Skrip dan Fungsi\Fungsi\bisect.sci');

```

Kemudian untuk fungsi yang dibuat secara inline dapat langsung digunakan tanpa harus menggunakan perintah `exec`.

Berikut ini adalah contoh penggunaan fungsi `bisect` dan `h` yang telah dibuat.

```

--> h(1)
ans =

-3.

```

```
--> x0 = bisect(h,1,2,1e-6)
x0 =

    1.6717
```

Komputasi Numerik

Di dalam Scilab, penyelesaian berbagai macam persoalan dalam komputasi numerik dapat dilakukan dengan mudah, seperti yang ditunjukkan pada contoh-contoh di bawah ini.

Sistem persamaan linear $Ax = b$ dapat diselesaikan dengan mudah menggunakan operator pembagian kiri (\backslash). Notasi $A \backslash b$ adalah ekuivalen dengan $\text{inv}(A)*b$.

```
--> A = [1 2 1 4; 2 0 4 3; 4 2 2 1; -3 1 3 2]
A =
```

```
    1.    2.    1.    4.
    2.    0.    4.    3.
    4.    2.    2.    1.
   - 3.    1.    3.    2.
```

```
--> b = [13; 28; 20; 6]
b =
```

```
    13.
    28.
    20.
     6.
```

```
--> x = A\b
x =
```

```
     3.
    - 1.
     4.
     2.
```

Nilai determinan dan inverse suatu matrik A dapat dihitung dengan menggunakan fungsi $\text{det}(A)$ dan $\text{inv}(A)$.

```
--> det(A)
ans =
```

```
- 180.
```

```
--> inv(A)
ans =
```

```
    0.    0.0833333    0.0833333   -0.1666667
    0.0666667   -0.3444444    0.3222222    0.2222222
   -0.2    0.1166667    0.1166667    0.1666667
    0.2666667    0.1222222   -0.2111111   -0.1111111
```

Penyelesaian suatu persamaan nonlinier $f(x) = 0$ dapat dilakukan dengan fungsi fsolve , seperti yang ditunjukkan pada contoh di bawah ini.

```
--> function y = h(x)
>   y = x^3 - x - 3
> endfunction
```

```
--> [x0, h0] = fsolve(1,h)
h0 =
- 8.882D-16
x0 =
1.6716999
```

Diperoleh bahwa akar dari $f(x) = x^3 - x - 3 = 0$, adalah $x = 1.6717$.

Terdapat banyak persamaan integral tertentu yang sulit atau bahkan tidak dapat diselesaikan secara analitis, salah satunya adalah seperti yang terdapat pada contoh berikut ini.

$$Q = \int_0^5 \frac{\sin(3x)}{\sqrt{x^2 + x + 1}} dx$$

Di dalam Scilab, nilai suatu integral tertentu dapat dihitung dengan menggunakan fungsi `intg`. Misalkan $h(x)$ adalah fungsi yang diintegrasikan pada persamaan di atas:

```
--> function y = h(x)
> y = sin(3*x)/sqrt(x^2 + x + 1)
> endfunction
```

Selanjutnya, nilai integral dapat dihitung dengan perintah sebagai berikut.

```
--> Q = intg(0, 5, h)
Q =
0.3648728
```

Jawaban yang diperoleh yaitu $Q = 0.36487$.

Daftar Pustaka

- Baudin, M. 2011. Introduction to Scilab. The Scilab Consortium - Digiteo.
Baudin, M. 2011. Programming in Scilab. The Scilab Consortium - Digiteo.
Campbell, S.L, Chancelier, J-P, Nikoukhah, R. 2010. Modeling and Simulation in Scilab/Scicos with Scicoslab 4.4 2ed. Springer. New York.
Chapra, S.C., Canale, R.P. 2015. Numerical Methods for Engineers 7ed. McGraw-Hill. New York.
Gomez, C., dkk. 1999. Engineering and Scientific Computing with Scilab. Birkhauser. Boston.
Heikell, J. 2011. Scilab for Real Dummies, Introducing an Open-Source Alternative to Matlab. Espoo.