

Aim

To learn programming for simple functions in SCILAB environment

Theory

SCILAB provides a built-in editor called SCIPAD/SCINOTES, wherein you can type the code for functions and compile and load them into the workspace. Clicking on **Applications** in the tool bar and selecting the **editor** invoke SCIPAD.

The syntax for writing programs in SCIPAD is as follows:

```
function [output arguments] = functionname (input arguments)
    instructions
endfunction
```

In the above code, `function` and `endfunction` are SCILAB keywords and must be exactly typed as shown this signifies start and end of a function definition.

1. Write a simple code for adding to numbers say (a + b) using the above-mentioned syntax.

- a. Open the SCIPAD/SCINOTES and type the following function line

```
function [plus] = pmf(a, b)
    plus =a+b
endfunction
```

- b. Execute the function by clicking play button (tab) in the SCIPAD window.
You will find a typical executing syntax invoked in the workspace as

```
-->exec('/Users/admin/pmf.sci', -1)
```

- c. now type the variable values (a +b) by calling the function name in the command prompt of workspace window as given below:

```
-->pmf(1, 2)
```

ans =

2. Write a program (or code) to calculate the length of a line in the x - y plane, given the coordinates of its two ends (x_1, y_1) and (x_2, y_2) . The length of a line is given by the relation $l = \sqrt{(x_1 - x_2)^2 + (y_2 - y_1)^2}$.

Open the SCIPAD/SCINOTES and type the following function

```
function [dist]=pmf (x1, x2, y1, y2)
dist= sqrt ((x1-x2) ^2+(y2-y1) ^2)
endfunction
```

Goto console window call the function and execute as given below:

```
-->pmf (23, 45)
```

ans =

68.

```
-->pmf (1, 2)
```

ans =

3.

3. Write a program to convert the trigonometric values from radians to degrees. That is the input values in the code is in the radians and output of the code should produce the values in degrees. Hint: degrees = radians (180/3.14).

Open the SCIPAD/SCINOTES and type the following function

```
function [deg]=pmf (a)
deg=(a.*180/%pi)
endfunction
```

The output is given by

```
-->pmf (%pi/6)
```

ans =

30.

Goto console window call the function and execute as given below:

```
-->pmf (%pi/6)
```

ans =

30.

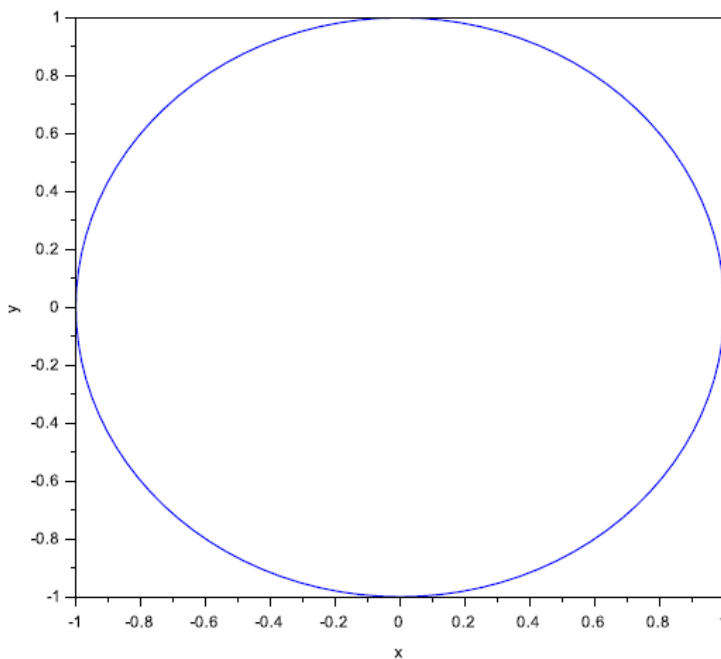
4. Write a function file (or code) to produce a circle. Hint. Use the problem statement given in the exercise no. 2 of tutorial 5.

Open the SCIPAD/SCINOTES and type the following function

```
function [circle]=pmf(x,y)
theta=[0:%pi/50:2.*%pi]'
    x=cos(theta);
    y=sin(theta);
plot(x,y);
xlabel('x');
ylabel('y');
endfunction
```

Goto console window call the function and execute as given below:

```
-->pmf(theta)
```



5. Write a function that produces an output conversion table for Celsius and Fahrenheit temperatures. The input the function should be two numbers T_i and T_f , specifying the lower and upper range of the table in Celsius. The output should be a two column matrix: the first column showing the temperature in Celsius from T_i to T_f in the increments of 1°C and the second column showing the corresponding temperatures in Fahrenheit. Hint: To do this, (i) create a column vector C from T_i to T_f with the command $C=[T_i:T_f]'$, (ii) calculate the corresponding numbers in Fahrenheit using the formulae $[F = 9/5C + 32]$, and (iii) make the final matrix

with the command `temp = [C F];` . Note that your output variable will be named `temp`.

Open the SCIPAD/SCINOTES and type the following function

```
function [F]=pmf(C)
    C=[ti:tf]'
    F = C.*9/5 + 32
Endfunction
```

Goto console window call the function and execute as given below:

```
->temp=[C pmf(20)]
temp =
```

```
20.    68.
21.    69.8
22.    71.6
23.    73.4
24.    75.2
25.    77.
26.    78.8
27.    80.6
28.    82.4
29.    84.2
30.    86.
31.    87.8
32.    89.6
33.    91.4
34.    93.2
35.    95.
36.    96.8
37.    98.6
38.   100.4
39.   102.2
40.   104.
```

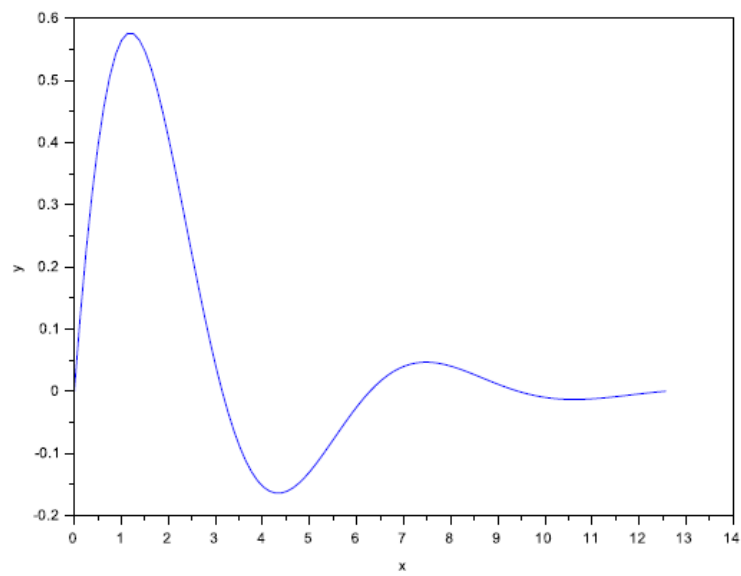
6. Write a code for an exponentially decaying sine plot as given in problem no. 5 of tutorial 5.

Open the SCIPAD/SCINOTES and type the following function

```
function [y]=pmf(x)
    x=[0:%pi/25:4.*%pi]'
    y=exp(-0.4.*x).*sin(x);
plot(x,y);
xlabel('x');
ylabel('y');
endfunction
```

Goto console window call the function and execute as given below:

-->pmf (x)



Result

Thus we learned the SCILAB programming for simple relations.