

Universidad del Valle
Escuela de Ingeniería de Sistemas y Computación
Curso: Métodos Numéricos
Docente: Daniel Barragán Calderón

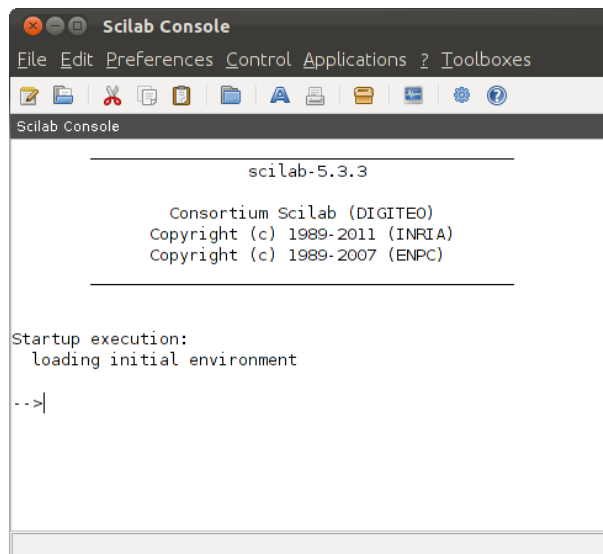
Clase Introducción a la Programación con Scilab (Parte 3)

Introducción

Scilab fue creado en 1990 por investigadores en INRIA (*Institut national de recherche en informatique et en automatique*) y ENPC (*École nationale des ponts et chaussées*). Scilab es un lenguaje de programación de código abierto, multiplataforma, orientado a cálculos numéricos. Puede ser usado para procesamiento de señales, análisis estadístico, tratamiento de imágenes, simulación de fluidos, optimización y modelamiento y simulación de sistemas dinámicos.

Enlace de descarga:

<http://www.scilab.org/>



Interfaz Gráfica Scilab

Guía Básica

En esta guía se abordaran otros aspectos de la programación y visualización de funciones con Scilab.

Funciones (Continuación)

Scilab permite crear funciones que reciben otras funciones como parámetro. Las funciones que se reciben como parámetro pueden ser evaluadas al interior de la función que las recibe.

Digite las siguientes instrucciones en un script de nombre ***evaluar1*** y almacenelo en la carpeta ***tallerscilab***. Ejecute el script por medio del botón del triangulo o por medio del menú ***Execute***, con la opción ***load into Scilab***.

```
function resultado=evaluar1(funcion, x, y)
    resultado = funcion(x, y)
endfunction
```

Digite las siguientes instrucciones en un script de nombre ***prueba1*** y almacenelo en la carpeta ***tallerscilab***. Ejecute el script.

```
function fx=funcion(x, y)
    fx = x + y;
endfunction

x = 100
y = 150
resultado = evaluar1(funcion, x, y)
```

La función evaluar recibe como parámetros: una función, un valor de x y un valor de y. La función evaluar evalúa la función que recibe como parámetro con los valores de x y y. Verifique que el resultado de la ejecución es resultado = 250.

Digite las siguientes instrucciones en un script de nombre ***evaluar2*** y almacenelo en la carpeta ***tallerscilab***. Ejecute el script por medio del botón del triangulo o por medio del menú ***Execute***, con la opción ***load into Scilab***.

```
function resultado=evaluar2(funcion, x)
    resultado = funcion(x)
endfunction
```

Digite las siguientes instrucciones en un script de nombre ***prueba2*** y almacenelo en la carpeta ***tallerscilab***. Ejecute el script.

```
function fx=funcion(x)
    v = 36, t = 4, cd = 0.25, g = 9.81;
    fx = sqrt(g*x/cd).*tanh(sqrt(g*cd./x)*t) - v;
endfunction
```

```
x = 145
resultado = evaluar2(funcion, x)
```

La función evaluar recibe como parámetros: una función y un valor de x. La función evaluar evalúa la función que recibe como parámetro con el valor de x. Verifique que el resultado de la ejecución es resultado = 0.0456265 .

Scilab permite la definición de funciones por medio del comando **deff**. Digite las siguientes instrucciones en un script de nombre **definicion1** y almacenelo en la carpeta **tallerscilab**. Ejecute el script.

```
deff('z=f(x,y)','z=x^4-y^4')
x= -3:0.2:3 ; y=x ;
z=feval (x1, x2, f) ;
```

Las instrucciones anteriores permiten evaluar la función especificada por medio del comando **deff** para los valores de x y y. El valor de z se obtiene por medio de la función **feval**. Tenga en cuenta que z debe ser una matriz.

Scilab permite la declaración de polinomios por medio del comando **poly**. Digite las siguientes instrucciones en un script de nombre **definicion2** y y almacenelo en la carpeta **tallerscilab**. Ejecute el script.

```
P1=poly([1 2 3],"x","coeff")
P2=poly([1 2 3],"x","roots")
x=(-5:1:5);
y1 = horner(P1,x)
y2 = horner(P2,x)
```

Las instrucciones anteriores muestran como declarar un polinomio de dos maneras, una de ellas por medio de los coeficientes del polinomio con el parámetro **coeff** y la otra por medio de las raíces del polinomio con el parámetro **roots**. El comando **horner** permite evaluar un polinomio sobre un arreglo de números u otros polinomios.

Problemas

1. Realice tres ejemplos de funciones que reciban una función y retornen el valor de la función evaluada.
2. Realice tres ejemplos donde defina funciones por medio de la función **deff** y evalúe las mismas por medio de la función **feval**.
3. Realice un ejemplo de declaración de un polinomio empleando el parámetro **coeff** y un ejemplo empleando el parámetro **roots**. Por medio de alguno de los algoritmos vistos en clase para el cálculo de las raíces encuentre las raíces del polinomio generado con el parámetro **coeff**.