Cálculo Numérico 2014
Trabajo Práctico 0

Introducción a Octave / Scilab

Trabajos de Laboratorio en Cálculo Numérico 2014: En el presente dictado de la materia, se propone la utilización de los software libres para cálculo científico `Scilab` u `Octave`. Los trabajos prácticos y demás actividades de Laboratorio estarán específicamente orientadas al uso de dichos programas, con el objetivo de que el alumno aprenda a utilizarlos como herramienta para desarrollar a nivel práctico los contenidos de la asignatura.

IMPORTANTE!!! Por otro lado, la presentación de los ejercicios que se deben entregar se realizará exclusivamente a través de la plataforma de la materia en http://www.cimec.org.ar/cursos/. Allí se indicará la fecha límite de entrega de cada práctico. Luego del vencimiento, no será posible subir la tarea a la página y se la considerará desaprobada. **El formato de los archivos será exclusivamente el pdf** y **no se aceptarán partes (gráficas, deducciones de fórmulas, etc.) escaneadas**.

Introducción: Tanto `Scilab` como `Octave` presentan las siguientes características: poseen una amplia variedad de librerías de funciones orientadas al cálculo científico, son interactivos, programables, de libre uso con la condición de hacer referencia a sus autores y disponibles tanto para plataformas Windows y Linux. El sitio oficial de `Scilab` es http://www.scilab.org y el de `Octave` es http://www.octave.org. Allí se encuentra información general, manuales, FAQs (*Frequently Asked Questions*), referencias sobre reportes, diferencias con Matlab, lista de errores, etc. y se pueden obtener la versiones binarias o los fuentes para las diferentes plataformas.

Ejercicio 1: Realice las siguientes operaciones utilizando funciones apropiadas de Scilab / Octave:

(a) $5^2 - \frac{1}{2^3} - \sqrt{3^2 + (2 \times 2)^2} =$

(b) $\sin(\pi/6) - \arctan(0.5) =$

(c) $\ln(3 + \frac{1}{5}) - e^2 =$

Nota: En la línea de comandos de Scilab / Octave, se pueden recuperar instrucciones ejecutadas anteriormente pulsando la flecha dirigida hacia arriba, lo que evita la escritura reiterada de una misma instrucción.

Ejercicio 2: Sea la función

$$y = \frac{\sin(2x)}{x(x+1)}$$

Halle el valor numérico de y, para $x = -4$, $x = -\pi/8$, $x = \sqrt{2}/4$, $x = \pi/2$ y $x = 9\pi/5$. Saque conclusiones sobre el dominio de la función. Se podrán calcular los valores de y correspondientes a $x = 0$ y a $x = -1$? Intente calcularlos y justifique su respuesta.

Ejercicio 3: Los siguientes ejemplos definen diferentes tipos de arreglos. Pruébelos y saque conclusiones:

(a) `[1 2 3 -4]`

(b) `[1 2 3 -4]'`

(c) `-2.5:0.5:1`

(d) `(-2.5:0.5:1)'`

(e) `[-3:2:4]`

Nota: Cuando las operaciones aritméticas $+$, $-$, $*$ y $/$ se utilizan entre matrices (donde un vector columna se puede interpretar como una matriz de $n \times 1$), debe tenerse en cuenta la compatibilidad de las dimensiones de las mismas, para que tales operaciones tengan sentido. Se presentan a continuación las operaciones correspondientes a la multiplicación, división y potenciación elemento a elemento. De esta manera, dadas dos matrices A con elementos A_{ij} y B con elementos B_{ij} se tiene que

(a) `A.*B` da como resultado una matriz `C` cuyos elementos son $C_{ij} = A_{ij} \cdot B_{ij}$

(b) `A./B` da como resultado una matriz `D` cuyos elementos son $D_{ij} = A_{ij}/B_{ij}$

(c) `A.^n` resulta ser otra matriz cuyos elementos son A_{ij}^n

Ejercicio 4: Considere los arreglos

$$x = \begin{pmatrix} -0.5 & 7 & 2 \end{pmatrix} \qquad \text{y} \qquad z = \begin{pmatrix} 2 \\ -1 \\ 3 \end{pmatrix}$$

Investigue qué realizan las siguientes operaciones:

(a) `2*x`

(b) `y = x-1`

(c) `x.*y`

(d) `x./y`

(e) `y.^2`

(f) `x'+5*z`

(g) `x+5*z`

(h) `y-z'/3`

(i) `v = [x,y]`

(j) `v(2:5)`

(k) `v(5:6)+(z(1:2))'`

(l) `w = [x;y]`

Ejercicio 5: Un polinomio se puede definir de dos maneras: por sus coeficientes o por sus raíces. Es necesario además indicar la variable simbólica para el polinomio. Investigue que es lo que realizan las siguientes intrucciones:

(a) `p = poly([2 3 5 7], 'x', 'coeff')`

(b) `q = poly([2 3 5], 'x', 'roots')`

(c) `roots(p)`

(d) `c = coeff(q)`

Ejercicio 6: Los siguientes ejemplos definen diferentes tipos de matrices. Pruebe y saque conclusiones:

(a) `A = [1 2 3;4 5 6;7 8 9]`

(b) `B = A'`

(c) `C = [-3.2,5,7.4,6;4,17,-1.3,2.1;5.9,-6,0,4.5]`

(d) `mat=C'`

(e) `C(1:2,2:4)`

(f) `C(:,3)`

(g) `C(2,:)`

(h) `zeros(5,2)`

(i) `ones(2,3)`

(j) `v = diag(A)`

(k) `D = diag(v,1)`

(l) `E = diag(v,-1)`

(m) `F = diag(5*ones(3,1),0)+diag(ones(2,1),-1)+diag(-3*ones(2,1),1)`

Ejercicio 7: Los siguientes comandos ejemplifican algunas de las posibilidades de manipulación de vectores que ofrecen Scilab / Octave. Trate de deducir que realiza en cada paso.

```
m = 5;
n = 4*m+1;
x = linspace(0,1,n);
y = zeros(1,n);
a = x(1:m+1);
y(1:m+1) = sin(2*%pi*a);
y(2*m+1:-1:m+2) = y(1:m);
y(2*m+2:n) = -y(2:2*m);
```

Ejercicio 8: Utilice los vectores x e y del punto anterior y grafíquelos con el comando `plot(x,y)`. A continuación grafique la siguiente función en el intervalo $[0, 2]$

$$f(x) = \left(\frac{x+1}{x^2+x+1}\right)^6 \cdot (\cos(x) + 3)$$

Ejercicio 9: Si A es una matriz cuadrada e invertible, el sistema $Ax = b$ tiene, teóricamente, una única solución. Investigue para un mismo sistema las siguientes instrucciones:

(a) `x1 = inv(A)*b`

(b) `x2 = A\b`

Ejercicio 10: En Scilab / Octave hay dos tipos de programas: los scripts y las funciones. Un script es simplemente una secuencia de órdenes. No tiene argumentos de entrada ni de salida. En cambio una función sí los tiene. Por otro lado, las variables definidas en un script son globales mientras que en una función, las variables definidas en la misma son locales.

Scripts: generalmente tienen la extensión `.sce` en `Scilab` (aunque no es obligatorio) mientras que en `Octave` tienen la extensión `.m`. El script puede estar colocado en cualquier carpeta. Puede escribirse en un editor cualquiera (por ej.: gnu emacs http://www.gnu.org/software/emacs/) o en Scipad para el caso de Scilab. Escriba un script con el siguiente contenido:

```
n = 100;
A = rand(n,n);
x0 = rand(n,1);
b = A*x0;
x = A\b;
```

En `Scilab` el script se ejecuta introduciendo la orden `exec prueba.sce` en la línea de comandos o mediante la barra de menú con la opción `exec` seleccionando el archivo correspondiente de la lista. Si se ejecuta desde Scipad, también mediante la barra de menú, eligiendo primero `Execute` y luego `Load into Scilab`. En `Octave` el script se ejecuta escribiendo el nombre del archivo (sin la extensión `.m`) en la línea de comandos.

Funciones: en `Scilab`, generalmente los nombres de los archivos de funciones tienen la extensión `.sci` mientras que en `Octave` la extensión es `.m` (como en los scripts). El esquema general de una función es

```
function [res1, res2, ...] = nombrefuncion(par1, par2, ...)
... ...
endfunction
```

donde `nombrefuncion` es el nombre de la función, que en el caso de `Octave` debe coincidir con el nombre del archivo donde está escrita. `part1`, `part2`, etc. son los argumentos de entrada y `res1`, `res2`, etc. son los argumentos de salida de la función. Escriba uno o varios archivos (según corresponda) con las siguientes funciones

```
function [x, y] = polarCart(r, t)
// Conversion de coordenadas polares a cartesianas.
x = r*cos(t) y = r*sin(t)
endfunction
//------------------------------------------------------
function [x, y] = polarCartGr(r, t)
// Conversion de coordenadas polares a cartesianas,
// el angulo esta dado en grados.
[x, y] = polarCart(r, t*%pi/180)
endfunction
//------------------------------------------------------
function fx = f(x)
fx = x(1)^2 + x(2)^2
endfunction
```

Los archivos de funciones se deben cargar en Scilab mediante alguna de las siguientes líneas de comando: `getf('func.sci');` `exec('func.sci');` o desde la barra de menú. También se pueden cargar desde Scipad, siguiendo el mismo procedimiento que para los scripts. En el caso de `Octave` no hace falta cargar la función en el entorno, basta con colocar el archivo en un lugar donde `Octave` pueda "verlo"(ver los comandos `path`, `addpath` y relacionados). Una vez cargada, la función se puede utilizar como las funciones *built-in* de Scilab / Octave. Pruebe las siguientes instrucciones y saque conclusiones:

```
[x1, y1] = polarCart(2, 0.7854)
[u, v] = polarCartGr(3, 30)

valor = f([3; 4])
x = [5; 6], res = f(x)
```

Ejercicio 11: Escriba una función de Scilab / Octave que calcule la fórmula de Baskara, ingresando sólo un vector con los coeficientes del polinomio cuadrático y obteniendo como salida no sólo las raíces sino también una leyenda indicando el tipo de raíz.