Cálculo Numérico 2025
Trabajo Práctico 0

Introducción a Octave

Trabajos de Laboratorio en Cálculo Numérico 2025: En el presente dictado de la materia, se utilizará *exclusivamente* el software para cálculo científico `Octave`. Los trabajos prácticos y demás actividades de Laboratorio estarán específicamente orientadas al uso de dicho programa, con el objetivo de que el alumno aprenda a utilizarlo como herramienta para desarrollar a nivel práctico los contenidos de la asignatura. Además, las evaluaciones parciales involucran el conocimiento de dicho lenguaje para poder resolver los ejercicios de programación.

IMPORTANTE: MODALIDAD DE LAS PRÁCTICAS

- **PLATAFORMA**: http://e-fich.unl.edu.ar/moodle.
  En ella encontrarán las guías de ejercicios, las presentaciones en diapositivas de la teoría y otros materiales de interés.
- **PRÁCTICOS DE EVALUACIÓN CONTINUA**: En fechas a convenir, se realizarán trabajos prácticos que consistirán en resolver un problema propuesto a través del aula virtual utilizando los conocimientos previamente adquiridos.
- **FECHA DE TRABAJOS PRÁCTICOS**: La fecha se indicará desde la cátedra. En caso de que el alumno no lo realice, se considera el práctico desaprobado.

Introducción: El software científico `Octave` presenta las siguientes características: posee una amplia variedad de librerías de funciones orientadas al cálculo científico, es interactivo, programable, de libre uso con la condición de hacer referencia a sus autores y disponible tanto para plataformas Windows y Linux. El sitio oficial de `Octave` es http://www.octave.org. Allí se encuentra información general, manuales, FAQs (*Frequently Asked Questions*), referencias sobre reportes, diferencias y similitudes 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 las funciones apropiadas de 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 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: En Octave los polinomios se representan por un vector de coeficientes ordenados de la mayor potencia hacia la menor. Por ejemplo, el polinomio $2x^3 - 3x^2 + 2$ se representa por el vector `[2, -3, 0, 2]`.
Sea `p = [2 3 5 7]` un vector que define los coeficientes de un polinomio. Investigue qué es lo que realizan las siguientes instrucciones:

(a) `polyout(p, 'x')`

(b) `polyval(p, x)` (Defina antes la variable x asignándole algún valor escalar y también como vector).

(c) `roots(p)`

Ejercicio 6 (Aula): 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 ofrece 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 (Aula): 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 (Aula): En 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: Un *script* es una secuencia de comandos que deseamos ejecutar a menudo y que por lo tanto nos gustaría no tener que escribirla cada vez que tenemos que usarla. Para esto podemos guardar la lista de comandos en un archivo de extensión `.m` y así poder ejecutarla tantas veces como queramos. El script puede estar colocado en cualquier carpeta y puede escribirse en un editor de textos cualquiera. Para mayor practicidad, se utiliza el procesador de texto incorporado en las nuevas versiones de `Octave`.

Escriba un script con el siguiente contenido:

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

En `Octave` el script se ejecuta escribiendo el nombre del archivo (sin la extensión `.m`) en la línea de comandos.

Funciones: 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.* `par1`, `par2`, 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
```

En `Octave`, no hace falta cargar la función en el entorno, basta con colocar el archivo en un lugar donde `Octave` pueda "verla"(ver los comandos `path`, `addpath` y relacionados). Una vez cargada, la función se puede utilizar como las funciones *built-in* de 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)
```

Una diferencia importante entre función y script, es que las variables definidas en una función sólo existen dentro de ella, mientras que en un script, una vez ejecutado, las variables pasan a formar parte de nuestro espacio de trabajo.

Ejercicio 11 (Aula): Escriba una función de 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.

Gráficas: En `Octave`, el comando `plot` dibuja puntos en el plano generados al utilizar las componentes del primer vector introducido *versus* las componentes del segundo vector, como vimos en el Ejercicio 8.

Ejercicio 12 (Aula): Se utiliza el comando plot para representar los puntos $(-1, 3)$, $(0, 2)$, $(1.5, 2)$ y $(2, 0)$:

```
x=[-1 0 1.5 2]; y=[3 2 2 0]; plot(x,y)
```

Investigue la diferencia que se producen con los siguientes comandos:

(a) `plot(x,y,'o')`

(b) `plot(x,y,'*')`

(c) `plot(x,y,'r')`

(d) `plot(x,y,'g*-')`

Podemos ver que cada vez que usamos el comando `plot` se genera un nuevo gráfico que sustituye al anterior. Si queremos que aparezcan varias gráficas conjuntamente, hay varias opciones. Una de ellas es la siguiente:

Ejercicio 13: Comentar al lado de las siguientes instrucciones lo que realiza cada una:

```
x=linspace(0,2*pi,201);
y=sin(x);
plot(x,y)
hold on
z=cos(x);
plot(x,z,'k-.')
hold off
```