

Práctica 1 — Octave/Matlab I

Esta práctica abarca los siguientes temas:

- Algoritmos, programas y lenguajes. Datos e instrucciones. Computadoras analógicas y digitales. La arquitectura de von Neumann. Almacenamiento primario y secundario: memoria principal, periféricos, dispositivos de entrada, de salida y de entrada/salida.
- El intérprete Octave/Matlab. Expresiones aritméticas. Variables escalares, matriciales y alfanuméricas. Algunas funciones para manipulación de vectores y matrices. Definición de funciones mediante la construcción de función anónima. Programas. Entrada y salida por consola. Estructuras de control: secuencia, decisión, iteración.

Bibliografía:

- J. W. Eaton, D. Bateman y S. Hauberg, *GNU Octave*.
- G. Borrell i Nogueras, *Introducción a Matlab y Octave*.

Verificación del entorno de trabajo. Para realizar esta práctica es necesario contar con un intérprete `Matlab` u `Octave` correctamente instalado. Antes de proceder verifique que puede hacer las siguientes operaciones:

- Ejecutar el intérprete.
- Ingresar expresiones aritméticas simples por el teclado y obtener una respuesta en pantalla.
- Crear un archivo de texto llamado `hola.m` que contenga la línea `printf("Hola\n");`
- Ejecutar ese archivo desde el intérprete (escribiendo simplemente `hola`).
- Utilizar la orden `help` para obtener información acerca de las funciones y órdenes, pruebe por ejemplo `help pi`.

Problema 1. Expresiones aritméticas.

- Evalúe las siguientes expresiones e interprete el resultado:

```
2*(4+21.1)
212/21
88^(1/3)
sqrt(-4)
log(10)
log(-10)
exp(pi*i)+1
```

- Obtenga el valor en radianes de un ángulo de 32° .
- Calcule

```
log(-1)
10^308
10^309
10^400
```

Problema 2. Variables. En Octave/Matlab, las variables pueden almacenar valores numéricos escalares, vectoriales o matriciales, o bien valores alfanuméricos (cadenas de caracteres). Para este ejercicio necesita saber asignar variables escalares y matriciales, extraer un elemento de matriz (mediante el uso de subíndices) y operar con matrices, distinguiendo las operaciones matriciales de las operaciones elemento a elemento. Asegúrese de comprender el significado de las sentencias que siguen y de poder predecir su resultado. Verifíquelo con el intérprete.

```
a=10;
v = [ 10 20 30 ];
A = [ 1 2 ; 3 a ];
B = [ v ; v ];
a*A;
A(1,2);
A*B;
C = [ a a ; 2*a sqrt(a) ];
A*C;
A.*C;
inv(A);
A./C;
dot(v,v);
r='producto';
```

También le serán útiles las funciones `linspace`, `sum`, `polyval`, `roots`, `who`.

- a) Asigne valores numéricos a tres variables escalares a , b y c . Asigne a d el valor de la expresión $\frac{a+\sqrt{3a+b^2}}{\frac{1}{c}-\frac{b}{c^a}}$.
- b) Calcule la suma de los primeros 1000 términos de la serie geométrica $\sum_{n=0}^{\infty} q^n$ con $q = 1/7$.

```
suma = 1.1666666666667
```

- c) Calcule las raíces de $x^3 + 2x^2 + 1$.

```
ans =
-2.20557 + 0.00000i
0.10278 + 0.66546i
0.10278 - 0.66546i
```

Evalúe luego el polinomio en las raíces obtenidas. ¿Qué resultado espera obtener? ¿Qué resultado se obtiene? ¿Puede explicar por qué?

- d) Obtenga la suma de los primeros 135 términos de la serie de la exponencial evaluados en $x = 2$.

```
Rta : 7.38905609893065040694182
```

Problema 3. Funciones simples. La forma más sencilla de definir una función que evalúa una expresión aritmética y devuelve el resultado es utilizando la notación llamada de “función anónima”:

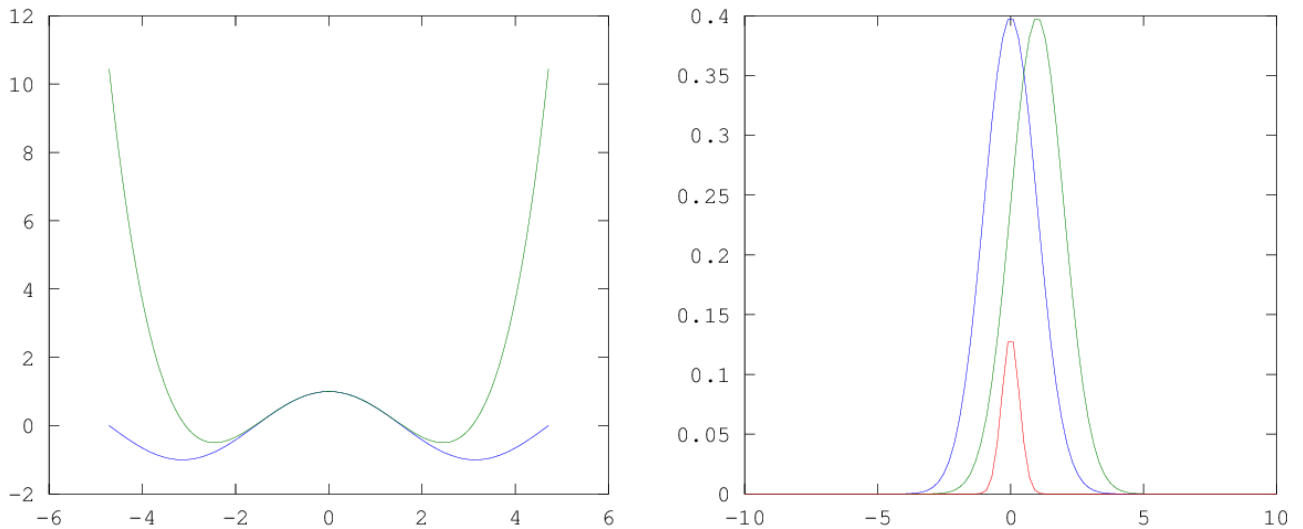
```
f = @(x) x.^2;
f(4)
f([1 2 3 4])

ans = 16
ans =

1    4    9   16
```

- a) Defina una función `ca(x)` que aproxime a la función coseno por su polinomio de Taylor de cuarto grado. Grafique el coseno y esa aproximación en el intervalo $[-3\pi/2, 3\pi/2]$.

b) Grafique la distribución gaussiana normalizada ($f(x) = e^{-(x-\mu)^2/2\sigma^2}/\sqrt{2\pi\sigma^2}$ para $\mu = 0, 1$ y $\sigma = 1, 3$).



c) Obtenga la integral de la función gaussiana con $\sigma = 1$ y $\mu = 0$ entre 0 y 1 (la función Octave/Matlab que calcula integrales numéricas se llama `quad`).

`ans = 0.34134`

Problema 4. Programas.

- Escriba un programa que convierta un ángulo de grados a radianes, leyendo el valor del ángulo por teclado.
- Escriba un programa que lea dos valores numéricos por teclado y los asigne a dos variables `x` e `y`. Luego de imprimir el contenido de las variables, deberá intercambiar sus valores y volverlas a imprimir.

Problema 5. Estructuras de control: ejecución condicional.

- Escriba un programa que lea dos números del teclado e imprima su cociente. Evite realizar una división por cero.
- Escriba un programa que devuelva el arco seno de un número ingresado por teclado. Advierta al usuario en caso de que la respuesta no exista en el conjunto de los números reales.
- Un automóvil parte del reposo con aceleración constante durante un intervalo de tiempo t_1 y luego continúa con velocidad constante, de modo que su posición en función del tiempo es

$$x(t) = \begin{cases} \frac{1}{2}at^2, & \text{si } 0 \leq t \leq t_1 \\ \frac{1}{2}at_1^2 + at_1(t - t_1), & \text{si } t > t_1. \end{cases}$$

Escriba un programa que lea el tiempo por teclado e imprima la posición correspondiente. Imprima un mensaje de error si el usuario ingresa un tiempo negativo.

Problema 6. Analice el siguiente fragmento de código y responda

- ¿Cuál de las tres ramas del `if` se ejecutará si el valor de `x` es 0?
- ¿Y si `x = 20`?
- ¿Qué valores toma y cuando se le asignan a `x` valores positivos muy grandes? Haga la prueba de asignar `x=Inf` y observe que valor toma `y`.

```

coef1 = 3 + log(1 + x^2);
coef2 = 25 * sqrt(x) * (1 + 4*x) / (3 + x*(x-1) );
if (x < coef1)
    y = 2*x - 3;
elseif (x < coef2)
    y = 1 / (1 + x);
else
    y = exp(-x/3);
end

```

Problema 7. Estructuras de control: iteraciones. Analice los siguientes bucles, en donde se hacen asignaciones a vectores:

```

for i = 1:2:10
    v(i) = 3*i;
    v(i+1) = -2*i;
end

```

```

j=1;
k=5;
while (j<k)
    u(j*k+1)=1;
    j=j+1;
    k=k-2;
end
printf('%d %d\n',j,k);

```

- a) ¿Qué valores toma i? ¿Cuáles son los elementos de v que son asignados, y con qué valores?
- b) ¿Qué valores toman los índices j y k? ¿Cuántas veces se ejecuta el bucle while?
- c) ¿Cuáles son los elementos de u que son asignados?
- d) Prediga el texto que imprime la sentencia printf.

Problema 8. Considere el siguiente fragmento de código y establezca a cuál de las expresiones indicadas corresponde.

```

f1 = x;
m1 = x*x;
kf1 = 1;
kf3 = 24;
suma = f1/6 + f1*m1/24;
for k = 2:20
    f1 = f1 * m1;
    kf1 = kf1 * k;
    kf3 = kf3 * (k+3);
    suma = suma + f1 / (kf1*kf3);
end

```

$$a) S = \sum_{k=0}^{20} \frac{x^{k+1}}{k!(k+3)!}$$

$$b) S = \sum_{k=0}^{20} \frac{x^{2k+1}}{k!(k+3)!}$$

$$c) S = \sum_{k=2}^{20} \frac{x^{k+1}}{k!(k+3)!}$$

$$d) S = \sum_{k=0}^{20} \frac{x^{2k+1}}{k(k+3)}$$

Problema 9. Escriba un programa que determine si un dado entero (leído por consola) es primo. Agregue un control para verificar que el número ingresado es un entero positivo. Le serán útiles las funciones mod(x,y), que calcula x módulo y, y fix(x), que devuelve su argumento sin la parte fraccionaria.