

# Computación

## Trabajo Práctico Nº 1: Introducción a Matlab/Octave

**Problema 1. Variables.** En Matlab u Octave, 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 una 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 de comandos.

Sentencia	Tipo de variable	Resultado
<code>a = 10;</code>		
<code>v = [ 10 20 30 ];</code>		
<code>A = [ 1 2 ; 3 a ];</code>		
<code>B = [ v ; v ];</code>		
<code>a * A;</code>		
<code>A(1,2);</code>		
<code>A * B;</code>		
<code>C = [a a ; 2*a sqrt(a)];</code>		
<code>A * C;</code>		
<code>A .* C;</code>		
<code>inv(A);</code>		
<code>A ./ C;</code>		
<code>dot (v,v);</code>		
<code>r = 'producto';</code>		

# Computación

## Trabajo Práctico Nº 1: Introducción a Matlab/Octave

**Problema 2.** Desarrolle los siguientes puntos utilizando las funciones `linspace`, `sum`, `polyval`, `roots`, `who`.

a) Asigne valores numéricos a tres variables escalares  $a$ ,  $b$  y  $c$ . Luego asigne a la variable  $d$  el valor de la expresión:

$$\frac{a + \sqrt{a + b^2}}{\frac{1}{c} - \frac{b}{c^a}}$$

b) Calcule la suma de los primeros 1000 términos, con  $q=1/7$ , de la serie geométrica:

$$\sum_{n=0}^{\infty} q^n$$

*Ayuda: dicho valor es: 1,16666666666667.*

c) Calcule las raíces de:

$$x^3 + x^2 + 1$$

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

*Ayuda: dichos valores son: -1.4656, 0.2328 + 0.7926i y 0.2328 - 0.7926i*

d) Obtenga la suma de los primeros 135 términos de la serie de Taylor de la función exponencial, alrededor de cero, evaluados en  $x = 2$ .

*Ayuda: dicho valor es: 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":

Ejemplo:

```
f = @(x) x.^2;
f(4)
ans = 16
f([1 2 3 4])
ans = 1 4 9 16
```

# Computación

## Trabajo Práctico N° 1: Introducción a Matlab/Octave

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

b) Grafique la distribución gaussiana normalizada:

$$f(x) = \frac{e^{-\frac{(x-\mu)^2}{2\sigma^2}}}{\sqrt{2\pi\sigma}}$$

Para  $\mu = 0$ , con  $\sigma = 1$  y  $\sigma = 3$ ; y luego para  $\mu = 1$ , con  $\sigma = 1$  y  $\sigma = 3$ .

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

*Ayuda: dicho valor es: 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:

# Computación

## Trabajo Práctico N° 1: Introducción a Matlab/Octave

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

Escriba un programa que lea el tiempo por teclado e imprima la posición correspondiente. Considere que el programa 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 la variable `y` cuando se le asignan a `x` valores positivos muy grandes? Haga la prueba de asignar `x=Inf` y observe qué valor toma ahora `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:

```
Bucle1
for i = 1:2:10
    v(i) = 3*i;
    v(i+1) = -2*i;
end

Bucle2
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);
```

# Computación

## Trabajo Práctico N° 1: Introducción a Matlab/Octave

- ¿Qué valores toma  $i$ ? ¿Cuáles son los elementos de  $v$  que son asignados, y con qué valores?
- ¿Qué valores toman los índices  $j$  y  $k$ ? ¿Cuántas veces se ejecuta el bucle `while`?
- ¿Cuáles son los elementos de  $u$  que son asignados?
- 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;  
f1=f1*m1;  
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.