



PRACTICO N° 6

Introducción

El objetivo de este práctico es practicar en la creación e invocación de scripts y funciones en Octave, mostrando las principales diferencias entre ambos.

Ejercicio 1

Considere los siguientes programas:

partel.m:

```
% script
a=a+2;
b=b+4;
c=a+b;
```

parte2.m:

```
% funcion
function [a,b]=parte2(x,y,z)
x=(y+z)/2;
c=x+y;
a=c;
b=y+z;
```

principal.m:

```
% principal
a=4;
b=6;
c=2;
partel;
a=c;
[c,b]=parte2(a,b,c);
```

Si ejecutamos desde la línea de comandos del Octave el script llamado *principal*, ¿cuáles son los valores finales de las variables *a*, *b* y *c*?

Ejercicio 2

Considere los siguientes programas:

medio.m:

```
%calcula pto medio
%delintervalo dado
% en vector x
a=x(1);
b=x(2);
pto = (a+b)/2;
```

extremos.m:

```
% calcula nuevos
% extremos
a=x(2);
b=x(1);
x(1)=(a+b)/2;
x(2)=4*(a+b)/2;
```

extremosF.m:

```
% calcula nuevos
% extremos
function x=extremosF(x)
a=x(2);
b=x(1);
x(1)=(a+b)/2;
x(2)=4*(a+b)/2;
```

Luego de copiar los programas ejecute desde la línea de comandos las siguientes instrucciones:

```
x=[3,15]
extremos;
medio;
pto
```

y ahora las siguientes:

```
x=[3,15]
extremosF(x);
medio;
pto
```

¿Cómo explica la diferencia encontrada al calcular el punto medio del intervalo?



Ejercicio 3

El siguiente algoritmo no se comporta como está especificado, encuentre el problema y corríjalo.

```
% Función mayor(x) devuelve el mayor de los elementos
% del vector x el cual puede estar desordenado
function maximo = mayor(x)
n=length(x);
maximo=x(1);
for i=2:n
    if x(i) > x(i-1)
        maximo = x(i);
    end
end
```

Ejercicio 4

- Escriba una función *agregarf* que agregue F nuevas filas a una matriz, rellenas con el número X.
- Escriba una función *agregarc* que agregue C nuevas columnas a una matriz, rellenas con el número X.
- Escriba una función *agregarfc* que agregue F nuevas filas y C nuevas columnas a una matriz, rellenas con el número X.

Ejercicio 5

Escriba una función *pertenece* que encuentre el número X en una matriz de MxN elementos. En caso de encontrarlo, la función debe retornar dos valores con la posición i, j de la primera aparición del elemento en la matriz. En caso de no encontrarlo, la función debe retornar ambos valores en cero.

Nota: la búsqueda se debe realizar por filas.

Ejercicio 6

Escribir la función *reverso* que toma como parámetro un vector v y devuelve un vector con los mismos elementos pero en orden inverso.

Ejemplo:

```
>> w = reverso([2 5 6 1 3 7])
w = [7 3 1 6 5 2]
```

Ejercicio 7

Se dispone de una función *en_intervalo* que recibe un número **valor** y dos números **inicio** y **fin**, devolviendo 1 si valor pertenece al intervalo [**inicio**,**fin**] y 0 en caso contrario.

Escriba una función *cantidad_en_intervalo* que reciba un vector **v_in** y dos números **inicio** y **fin**, y devuelva la cantidad de elementos de **v_in**, comprendidos en el intervalo [**inicio**,**fin**].

Sugerencia: Utilice la función *en_intervalo* dentro de la función *cantidad_en_intervalo*.

Ejercicio 8

Se dispone de una función *es_primo* que dado un entero devuelve 1 si el mismo es primo y 0 en caso contrario.

a) Escribir una función *listar_primos* que dados dos valores **a** y **b** devuelva un vector conteniendo los números primos comprendidos en el intervalo [**a**,**b**].

b) Escribir una función *expresar_como_suma* que dado un número entero **x** devuelva dos enteros primos **a** y **b** tal que $x = a + b$. De no existir a y b que cumplan tal propiedad, la función debe devolver a y b en -1.

Nota: Para la implementación en Octave se sugiere usar la función *isprime* que ya viene incluida en Octave.



Avanzados

Ejercicio 9

El programador del siguiente algoritmo cometió un error, identifíquelo y corríjalo. Este algoritmo ordena un vector de números de menor a mayor.

```
% ordena un vector

function x=ordenar(x)
n=length(x); %obtengo largo del vector
for i=1:n-1
    for j=i+1:n
        if x(i)>x(j) %intercambiar valores
            x(j)=x(i);
            x(i)=x(j);
        end
    end
end
end
```

Ejercicio 10

Escriba una función *traza* que calcule la suma de los elementos de la diagonal de una matriz de tamaño $N \times N$.

Ejercicio 11

Hay una sola opción correcta por cada pregunta

1. Indique cuál de las siguientes afirmaciones es verdadera:
 - a. en un *script* el alcance de las variables es local, mientras que en las funciones es global
 - b. tanto los *scripts* como las funciones tienen alcance global para sus variables
 - c. en un *script* el alcance de las variables es global, mientras que en las funciones es local
 - d. Ninguna respuesta es verdadera

1. ¿Cuál es el resultado del *script* de *Octave* del recuadro?
 - a. $x = 1, y = 5$
 - b. da error, porque está mal usada la sentencia de asignación
 - c. $x = 5, y = 4$
 - d. Ninguna respuesta es verdadera

$y = 1;$ $x = 4;$ $y + 1 = x;$

1. Dada la función del recuadro, indique qué algoritmo implementa:
 - a. $a * b$
 - b. $a * a$
 - c. a / b
 - d. Ninguno de los anteriores