

MATHEMATICA

Programación Procedural

Ricardo Villafaña Figueroa

Contenido

Funciones como Procedimientos	3
Ciclos	6
Ciclo Do.....	6
Ciclo For	8

FUNCIONES COMO PROCEDIMIENTOS

Ejemplo

Resolver el siguiente sistema de ecuaciones:

$$\begin{aligned} \text{eq1} &= 3x - 2y - 5 = 0; \\ \text{eq2} &= -4x + y - 4 = 0; \end{aligned}$$

Solución sin funciones especiales:

```
solucion = Solve[{eq1, eq2}, {x, y}]
```

$$\left\{ \left\{ x \rightarrow -\frac{13}{5}, y \rightarrow -\frac{32}{5} \right\} \right\}$$

Almacenar el par de resultados obtenidos en la variable par:

```
par = {x, y} /. solucion[[1]]
```

$$\left\{ -\frac{13}{5}, -\frac{32}{5} \right\}$$

Utilizando una función que integre los dos comandos anteriores.

Utilizando paréntesis como agrupador de comandos:

```
interseccion[ec1_, ec2_] :=
(
  solucion = Solve[{ec1, ec2}, {x, y}];
  par = {x, y} /. solucion[[1]];
  Return[par]
)
```

Ejemplo de uso:

```
interseccion[eq1, eq2]
```

```
{-13/5, -32/5}
```

Utilizando la función Module como agrupador de comando y definiendo variables locales:

```
interseccion[ec1_, ec2_] :=  
Module[{solucion, par}, (*variables locales*)  
  solucion = Solve[{ec1, ec2}, {x, y}];  
  par = {x, y} /. solucion[[1]];  
  Return[par]]
```

Ejemplo de uso:

```
interseccion[eq1, eq2]
```

```
{-13/5, -32/5}
```

Ejemplo

Definir una función que extraiga los coeficientes de una función lineal.

Solución

```
coeficientes[Linea_] :=
(
  a = Coefficient[Linea, x];
  b = Coefficient[Linea, y];
  c = Part[Linea, 1];
  Return[{a, b, c}]
)
```

Ejemplo de uso:

```
coeficientes[3 x - 2 y - 4]
```

```
{3, -2, -4}
```

Utilizando la función Module:

```
coeficientes[Linea_] :=
Module[{a, b, c}, (*variables locales*)
  a = Coefficient[Linea, x];
  b = Coefficient[Linea, y];
  c = Part[Linea, 1];
  Return[{a, b, c}]]
```

Ejemplo de uso:

```
coeficientes[x - 4 y - 6]
```

```
{1, -4, -6}
```

CICLOS

Ciclo Do

Ejemplo

Generar una tabla de conversión de grados a radianes, de cero a 90 grados con incremento de 5 unidades y una precisión de 8 cifras.

```
Do[Print[i, "\t", N[i  $\frac{\text{Pi}}$ , 8]], {i, 0, 90, 5}]
```

0	0
5	0.087266463
10	0.17453293
15	0.26179939
20	0.34906585
25	0.43633231
30	0.52359878
35	0.61086524
40	0.69813170
45	0.78539816
50	0.87266463
55	0.95993109
60	1.0471976
65	1.1344640
70	1.2217305
75	1.3089969
80	1.3962634
85	1.4835299
90	1.5707963

Ejemplo

Suma de los primeros n números enteros:

```
miSuma2[n_] :=  
Module[{s = 0},  
Do[s = s + i, {i, 1, n, 1}];  
s]
```

Probar la función:

```
miSuma2[5]
```

```
15
```

Ejemplo

Producto de los primeros n números enteros:

```
miProducto2[n_] :=  
Module[{p = 1},  
Do[p = p * i, {i, 1, n, 1}];  
p]
```

Probar la función

```
miProducto2[5]
```

```
120
```

Ciclo For

Suma de los primeros n números enteros:

```
miSuma1[n_] :=  
Module[{s = 0},  
  For[i = 1, i ≤ n, i++,  
    s = s + i];  
s]
```

Probar la función:

```
miSuma1[5]
```

```
15
```

Producto de los primeros n números enteros:

```
miProducto1[n_] :=  
Module[{p = 1},  
  For[i = 1, i ≤ n, i++,  
    p = p * i];  
p]
```

Probar la función

```
miProducto1[5]
```

```
120
```