

Presentación *resumen* del libro:

"EMPEZAR DE CERO A PROGRAMAR EN **lenguaje C**"

Autor: Carlos Javier Pes Rivas (correo@carlospes.com)

Capítulo 8

ASIGNACIÓN, SALIDA Y ENTRADA



OBJETIVOS

- Conocer las instrucciones primitivas, y saber hacer uso de ellas.
- Saber utilizar los elementos ya estudiados para escribir instrucciones primitivas.
 - De los elementos que se utilizan para diseñar algoritmos, sólo quedan por estudiar las instrucciones. En este capítulo se van a estudiar las tres instrucciones más elementales que existen en programación, conocidas como instrucciones primitivas (asignación, salida y entrada). Las instrucciones primitivas son las primeras que todo programador debe conocer, pero, ¿de qué manera están relacionadas con los elementos ya estudiados?

CONTENIDO

8.1 INTRODUCCIÓN

8.2 ASIGNACIÓN

8.3 SALIDA

8.4 ENTRADA

8.1 INTRODUCCIÓN

- En programación, las instrucciones que se utilizan para diseñar algoritmos se pueden clasificar en:
 - Primitivas.
 - De control.
 - Llamadas a subalgoritmos (llamadas a subprogramas).
- En este capítulo se van a explicar las instrucciones primitivas (existen tres):
 - Asignación.
 - Salida.
 - Entrada.

8.2 ASIGNACIÓN

- Una **instrucción de asignación** (o simplemente **asignación**) consiste en asignar el resultado de la evaluación de una expresión a una variable:

```
<nombre_de_la_variable> ↵ <expresión>
```

- **EJEMPLO:** Dadas las siguientes declaraciones de constantes y variables en pseudocódigo.

```
PI = 3.141592  
real area, longitud, radio = 5.78
```

Algunas instrucciones de asignación son:

```
area ↵ PI * radio ** 2  
longitud ↵ 2 * PI * radio
```

8.3 SALIDA (1/2)

- Una **instrucción de salida** (o simplemente **salida**) consiste en llevar hacia el exterior los valores (datos) obtenidos de la evaluación de una lista de expresiones:

```
escribir( <expresión_1>, <expresión_2>, ..., <expresión_n> )
```

También se puede escribir como:

```
escribir( <lista_de_expresiones> )
```

8.3 SALIDA (2/2)

- **EJEMPLO:** Partiendo de las variables:

```
cadena nombre = "Timoteo"  
entero edad = 27, hijos = 2
```

al escribir

```
escribir( nombre, " tiene ", edad, " años." )  
escribir( nombre, " tiene ", hijos, " hijos." )
```

por pantalla aparecerá:

```
Timoteo tiene 27 años.  
Timoteo tiene 2 hijos.
```

8.4 ENTRADA (1/3)

- Una **instrucción de entrada** (o simplemente **entrada**) consiste en asignar a una o más variables, uno o más valores (datos) recibidos desde el exterior:

```
leer( <nombre_de_la_variable_1> ,  
      <nombre_de_la_variable_2> ,  
      . . . ,  
      <nombre_de_la_variable_n> )
```

También se puede escribir como:

```
leer( <lista_de_variables> )
```

8.4 ENTRADA (2/3)

- **EJEMPLO:** Partiendo de las variables:

```
cadena nombre, apellidos  
entero edad
```

para cada una de ellas se puede recoger un valor (dato) desde el teclado, escribiendo:

```
leer( nombre )  
leer( apellidos )  
leer( edad )
```

Otra posibilidad es:

```
leer( nombre, apellidos, edad )
```

8.4 ENTRADA (3/3)

- **EJEMPLO:** Si se han declarado:

```
cadena nombre
real numero
```

al escribir:

```
escribir( "Introduzca su nombre: " )
leer( nombre )
escribir( "Introduzca un número real: " )
leer( numero )
escribir( nombre, ", el doble de ", numero, " es: ",
          numero * 2 )
```

por pantalla se verá
algo parecido a:

```
Introduzca su nombre: Juan
Introduzca un número real: 31.4
Juan, el doble de 31.4 es: 62.8
```

EJERCICIOS RECOMENDADOS

- **Resueltos:** 1, 2 y 3.
- **Propuestos:** 1, 2, 3 y 4.

GRACIAS POR SU ATENCIÓN

Para más información, puede visitar la web del autor:

<http://www.carlospes.com>

