

	Unitécnica Manizales Bootcamp Programación		
Programa	Bootcamp Programación	Grupo	Semana
Tema	Pseudocódigo		
Instructor	Julián Giraldo R.	Fecha	2025-08-04

PSEUDOCÓDIGO

La especificación de algoritmos mediante pseudocódigo, es una de las formas más utilizadas hoy en día. Su ventaja es que se hace en un lenguaje natural, similar a las representaciones mentales que podríamos tomar para resolver un problema. Originalmente este lenguaje se representó en idioma inglés, pero también hay una variante totalmente en español y por último, utiliza estructuras léxicas y sintácticas similares a las de cualquier lenguaje de programación imperativo.

Estructura de un algoritmo en pseudocódigo

Todo algoritmo cumple con una estructura sintáctica establecida, de manera que se puedan reconocer fácilmente sus partes y componentes. A continuación se presenta y explica la estructura de un algoritmo en pseudocódigo:

Algoritmo <i>nombre</i> Const <i>Declaración de constantes</i> Var <i>Declaración de variables</i> Inicio <i>Cuerpo de instrucciones del algoritmo el cual cumple normalmente con la estructura Entrada de datos, procesamiento y salida de resultados</i> Fin

Conceptos básicos de pseudocódigo

Comentarios: Los comentarios se refieren a todo texto que no debe hacer parte del programa y que se utiliza con fines informativos o explicativos únicamente. Un comentario puede ir en cualquier parte del algoritmo, y no debe interpretarse en ninguna situación. Un comentario sirve para explicar por ejemplo que hace un algoritmo, quien fue el creador, cuando lo hizo o que función cumple una variable. Para nuestro caso, utilizaremos dos tipos de comentarios:

Comentarios de línea: Este tipo de comentario se hace para solo una línea de texto, termina cuando se pasa a una nueva línea. Comienza con el símbolo // (barra inclinada doble) y no debe cerrarse con nada. Ejemplo.

```
Algoritmo xxxxx
// este es un comentario de linea
inicio
...
Fin
```

Comentarios de bloque: Este tipo de comentarios se hace para un bloque de texto, puede contener más de una línea, necesita símbolo de inicio y símbolo de fin de bloque. Se abre con /* y se cierra con */.

```
Algoritmo xxxxx
...
inicio
/*  este es un comentario de linea
    Tiene tres lineas comentadas
    Y todas hacen parte de un solo comentario */
...
Fin
```

Identificadores: Un identificador es un nombre que representa algún valor importante del programa. Con un identificador, se pueden nombrar algoritmos, variables, constantes simbólicas, arreglos, registros, funciones o procedimientos. Para construir identificadores pseudocódigo propone 1 regla obligatoria y 2 opcionales:

Reglas obligatorias:

1. Todo identificador debe iniciar por una letra y puede ir seguida por cero o más apariciones de números, letras o el carácter subrayado _

Reglas opcionales:

1. Un identificador debería construirse de forma que dé una idea aproximada del valor que está representando.
2. Un identificador no debería ser demasiado corto, ni extremadamente largo.

Ejemplos de identificadores

```
PI, valor1, sueldo, horas extras, hed, MAXIMO, suma
```

Aprender a formar y a reconocer identificadores bien formados es un conocimiento básico que todo estudiante de programación debe tener.

Tipos de datos: Cualquier valor que se utilice en un algoritmo, sea implícito o explícito, pertenece a un único conjunto de valores en particular. Es decir, los valores que se usan en el pseudocódigo tienen un sólo dominio de valores posibles. Estos dominios se conocen específicamente como tipos de datos. Los tipos de datos existentes en pseudocódigo son:

Categoría	Tipo	Definición	Rango de valores	Ejemplos
Numéricos	Entero	Número sin parte decimal	$(-\infty .. +\infty)$	-23; 0; 2547
	Real	Número con parte decimal, separada por un punto.	$(-\infty .. +\infty)$	-23.09; 1.2798; 0.54
Texto	Carácter	Un solo carácter entre comillas sencillas '.	a, b, ..., z, A, B, ..., Z, 0, 1, ..., 9, +, -, *, /, ^, >, =, <, ., ", ', (,), [,], /, ,	'a'; 'A'; 'Z'; '0'
	Cadena	Cualquier grupo de caracteres encerrado entre comillas dobles "".	"" cadena vacía unido con cualquier grupo de caracteres encerrados entre comillas dobles ""	""; "Hola mundo"; ""
Lógicos	Booleano	Uno de los valores Verdadero o Falso	{VERDADERO, FALSO}	VERDADERO FALSO

Constantes: Representa un valor que es conocido desde el inicio del algoritmo, no cambia durante la ejecución del mismo e inclusive, es probable que nunca cambie o se modifique sólo después de largos períodos de tiempo (meses, años). Las constantes pueden ser valores puntuales, que se conocen como **literales**, por ejemplo 12, "HOLA", 'A', 3.1416, etc. De otro lado, cuando una constante se usará muchas veces dentro del algoritmo o su valor puede cambiar de un período a otro, se acostumbra a usar identificadores para nombrar las variables.

Ej.: PI = 3.1416, E=1.2128, DIASMES=30, NOMBREEMPRESA="Los Marinillos", etc.

Para definir una constante nombrada, basta con escribir el nombre de esta (que por buena práctica de programación se acostumbra a escribir en mayúsculas sostenidas) el operador igual (=) y el valor que posee.

Variables: Las variables representan valores que pueden cambiar durante la ejecución de un algoritmo o programa. Un valor puede cambiar por que:

- Es el usuario quien lo asigna (variables de entrada)
- Guarda el resultado intermedio o definitivo de una operación (variables Internas y de salida)

Expresiones: Una expresión corresponde a una operación que implica operandos (los valores involucrados) y un operador (la forma de relacionar esos valores), existen diversas categorías de expresiones y de acuerdo con cada expresión, se pueden tener operadores y tipos de datos involucrados.