

Pseudocódigo y DFD

Desarrollo backend JAVA



Recuerda:

"La paciencia y la perseverancia tienen un efecto mágico ante el cual desaparecen las dificultades y desaparecen los obstáculos."



Para implementar la solución de un problema mediante el uso de una computadora es necesario establecer una serie de pasos que permitan resolver el problema, a este conjunto de pasos se le denomina algoritmo, el cual debe tener como característica final la posibilidad de transcribirlo fácilmente a un lenguaje de programación, para esto se utilizan herramientas de programación, las cuales son métodos que permiten la elaboración de algoritmos escritos en un lenguaje entendible.

Un algoritmo, aparte de tener como característica la facilidad para transcribirlo, debe ser:

- **Preciso.** Debe indicar el orden en el cual debe realizarse cada uno de los pasos que conducen a la solución del problema.
- **Definido.** Esto implica que el resultado nunca debe cambiar bajo las mismas condiciones del problema, éste siempre debe ser el mismo.
- **Finito.** No se debe caer en repeticiones de procesos de manera innecesaria; deberá terminar en algún momento.

Por consiguiente, el algoritmo es una serie de operaciones detalladas y no ambiguas para ejecutar paso a paso que conducen a la resolución de un problema, y se representan mediante una herramienta o técnica, o bien, es una forma de describir la solución de un problema planteado en forma adecuada y de manera genérica.

Además de esto, se debe considerar que el algoritmo, que posteriormente se transformará en un programa de computadora, debe considerar las siguientes partes:

- Una descripción de los datos que serán manipulados.



- Una descripción de acciones que deben ser ejecutadas para manipular los datos.
- Los resultados que se obtendrán por la manipulación de los datos.

Las herramientas o técnicas de programación que más se utilizan y que se emplearán para la representación de algoritmos a lo largo de esta guías son dos:

- **Pseudocódigo.**
- **Diagramas de flujo.**

IDENTIFICADORES

Antes de analizar cada una las herramientas que se utilizan en representación de algoritmos para la solución de problemas, se establecerá qué son los identificadores que se utilizan dentro de un algoritmo.

Los identificadores son los nombres que se les asignan a los objetos, los cuales se pueden considerar como variables o constantes, éstos intervienen en los procesos que se realizan para la solución de un problema, por consiguiente, es necesario establecer qué características tienen.

Para establecer los nombres de los identificadores se deben respetar ciertas reglas que establecen cada uno de los lenguajes de programación, para el caso que nos ocupa se establecen de forma indistinta según el problema que se esté abordando, sin seguir regla alguna, generalmente se utilizará la letra, o las letras, con la que inicia el nombre de la variable que representa el objeto que se va a identificar.

Constante

Un identificador se clasifica como constante cuando el valor que se le asigna a este identificador no cambia durante la ejecución o proceso de solución del problema. Por ejemplo, en problemas donde se utiliza el valor de **PI**, si el lenguaje que se utiliza para codificar el programa y ejecutarlo en la computadora no lo tiene definido, entonces se puede establecer de



forma constante estableciendo un identificador llamado PI y asignarle el valor correspondiente de la siguiente manera:

$$PI = 3.1416.$$

De igual forma, se puede asignar valores constantes para otros identificadores según las necesidades del algoritmo que se esté diseñando.

Variables

Los identificadores de tipo variable son todos aquellos objetos cuyo valor cambia durante la ejecución o proceso de solución del problema. Por ejemplo, el sueldo, el pago, el descuento, etcétera, que se deben calcular con un algoritmo determinado, o en su caso, contar con el largo (L) y ancho (A) de un rectángulo que servirán para calcular y obtener su área. Como se puede ver, tanto L como A son variables que se proporcionan para que el algoritmo pueda funcionar, y no necesariamente se calculan dentro del proceso de solución.

Tipos de variables

Los elementos que cambian durante la solución de un problema se denominan variables, se clasifican dependiendo de lo que deben representar en el algoritmo, por lo cual pueden ser: de tipo entero, real y string o de cadena, sin embargo, existen otros tipos de variables que son permitidos con base en el lenguaje de programación que se utilice para crear los programas, por consiguiente, al momento de estudiar algún lenguaje de programación en especial se deben dar a conocer esas clasificaciones.

Para el caso de esta guía, se denominará variables de tipo entero a todas aquellas cuyo valor no tenga valores decimales; contrario a las de tipo real, la cual podrá tomar valores con decimales. Como ejemplo de variables enteras se puede considerar el número de personas, días trabajados, edad de una persona, etcétera. Y para el caso de reales, se puede considerar el sueldo de una persona, el porcentaje de equis cantidad, etcétera.



En caso de que las variables tomen valores de caracteres, se designarán string o de cadena; como ejemplo de éstas se pueden mencionar el sexo de una persona, falso o verdadero, el nombre de una persona, el tipo de sangre, etcétera.

Tabla de Variables

En el desarrollo de un ciclo de vida de un sistema de información, se lleva a cabo un proceso de análisis y levantamiento de información para cumplir con la resolución del problema propuesto. En esta etapa se recopila todo lo que está disponible para el desarrollo y se describe qué cosas se necesitan desarrollar para lograr el objetivo.

En esta guía usaremos una herramienta que nos ayudará a detectar y definir las variables y constantes que se deben utilizar en el desarrollo de un programa: **tabla de variables**. Esta herramienta es simplemente una tabla que contiene 3 columnas:

- **Nombre:** indica cómo se llamará la variable a definir
- **Descripción:** muestra una breve descripción de esta variable
- **Tipo:** expresa que tipo de variable debe ser: entero, decimal, cadena de caracteres, entre otros

NOMBRE	DESCRIPCIÓN	TIPO
A	Altura del cubo	Entero
B	Base del cubo	Entero
C	Largo del cubo	Entero

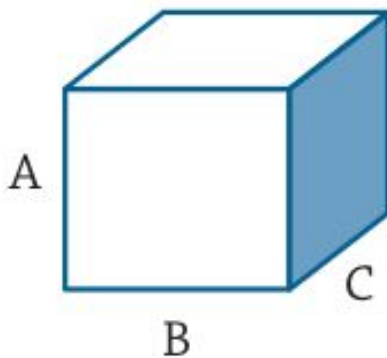


PSEUDOCÓDIGO

Sin duda, en el mundo de la programación el pseudocódigo es una de las herramientas más conocidas para el diseño de solución de problemas por computadora. Esta herramienta permite pasar casi de manera directa la solución del problema a un lenguaje de programación específico. El pseudocódigo es una serie de pasos bien detallados y claros que conducen a la resolución de un problema.

La facilidad de pasar casi de forma directa el pseudocódigo a la computadora ha dado como resultado que muchos programadores implementan de forma directa los programas en la computadora, cosa que no es muy recomendable, sobre todo cuando no se tiene la suficiente experiencia para tal aventura, pues se podrían tener errores propios de la poca experiencia acumulada con la solución de diferentes problemas.

Por ejemplo, el pseudocódigo para determinar el volumen de una caja de dimensiones A, B y C se puede establecer de la siguiente forma:



1. Inicio.
2. Leer las medidas A, B y C.
3. Realizar el producto de $A * B * C$ y guardarlo en V.
4. Escribir el resultado V.
5. Fin.

Como se puede ver, se establece de forma precisa la secuencia de los pasos por realizar; además, si se le proporciona siempre los mismos valores a las variables A, B y C, el resultado del volumen será el mismo y, por consiguiente, se cuenta con un final.



DIAGRAMAS DE FLUJO

Los diagramas de flujo son una herramienta que permite representar visualmente qué operaciones se requieren y en qué secuencia se deben efectuar para solucionar un problema dado. Por consiguiente, un diagrama de flujo es la representación gráfica mediante símbolos especiales, de los pasos o procedimientos de manera secuencial y lógica que se deben realizar para solucionar un problema dado.

Los diagramas de flujo desempeñan un papel vital en la programación de un problema, ya que facilitan la comprensión de problemas complicados y sobre todo aquellos en que sus procesos son muy largos; 2 generalmente, los diagramas de flujo se dibujan antes de comenzar a programar el código fuente, que se ingresará posteriormente a la computadora.

Los diagramas de flujo facilitan la comunicación entre los programadores y los usuarios, además de que permiten de una manera más rápida detectar los posibles errores de lógica que se presenten al implementar el algoritmo. En la tabla 1.1 se muestran algunos de los principales símbolos utilizados para construir un diagrama de flujo.

Dentro de los diagramas de flujo se pueden utilizar los símbolos que se presentan en la tabla 1.2, con los cuales se indican las operaciones que se efectuarán a los datos con el fin de producir un resultado.







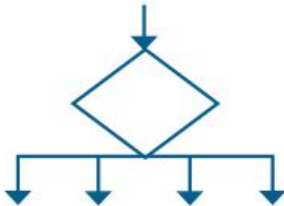



Símbolo	Significado
	Terminal /Inicio.
	Entrada de datos.
	Proceso.
	Decisión.
	Decisión múltiple.
	Imprimir resultados.
	Flujo de datos.
	Conectores.

Tabla 1.1 Principales símbolos utilizados para construir los diagramas de flujo.



Símbolo	Operación
+	Suma
-	Resta
*	Multiplicación
/	División
^	Exponenciación
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
< >	Diferente que
=	Igual que

Tabla 1.2 Principales símbolos utilizados en los diagramas de flujo para indicar las operaciones que se realizan para producir un resultado.



Por ejemplo, se puede establecer la solución del diagrama de flujo para determinar el volumen de una caja de dimensiones A, B y C como se muestra en la figura 1.1.

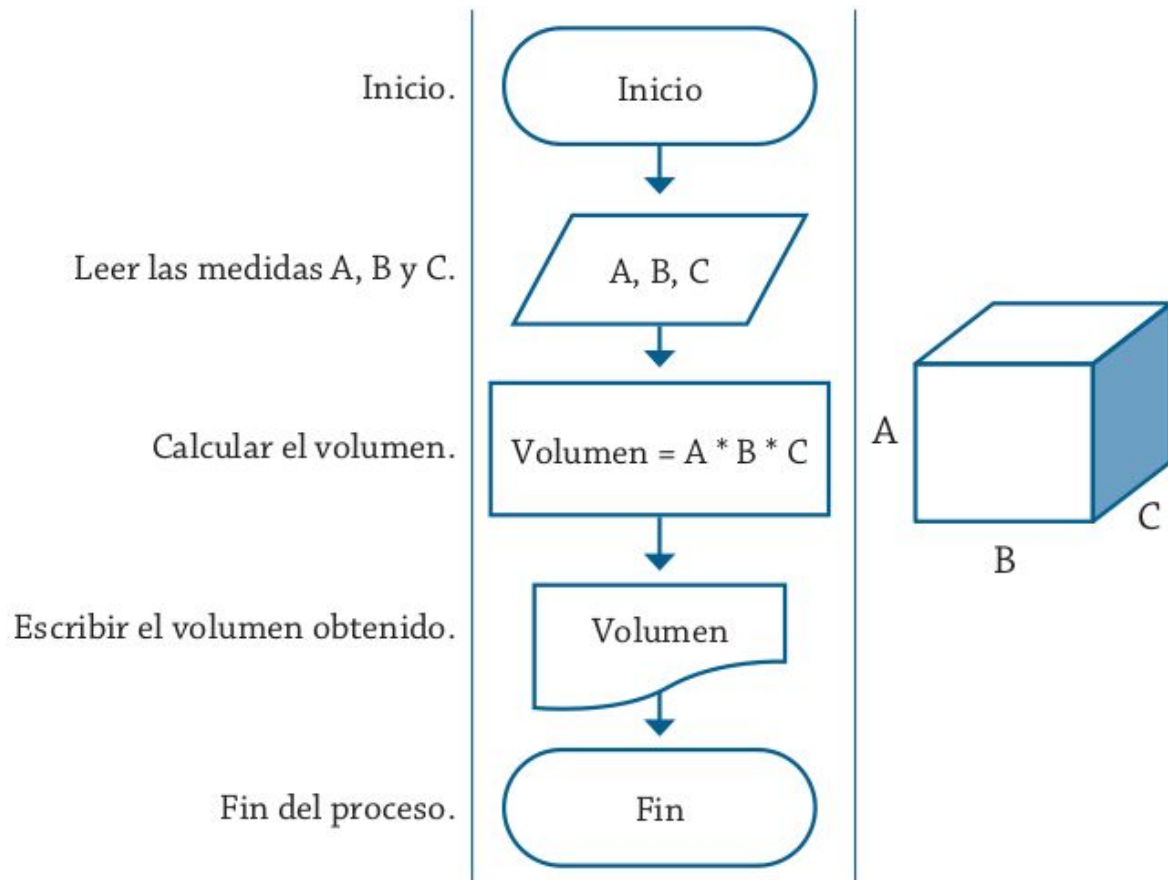


Figura 1.1 Diagrama de flujo para obtener el volumen de un cubo.

