



# Nociones Fundamentales

**Este capítulo ayuda al lector a entender y tomar contacto con los conceptos elementales de la programación. A partir de estos se podrá iniciar el aprendizaje de la programación y la destreza en el diseño de algoritmos básicos.**

## 1.1 Proceso

El primer concepto que debe tener claro un estudiante de programación es lo que significa un PROCESO. Se puede definir PROCESO de varias formas:

- Acción que se puede descomponer en otras más simples
- Conjunto de fenómenos organizados en el tiempo
- Conjunto de acciones elementales que forman un acontecimiento

Todas dicen casi lo mismo de diferentes formas. En todas las actividades que desarrolla el ser humano podemos encontrar procesos, por ejemplo, en el Proceso de Admisión por el que han pasado los alumnos que han ingresado al Instituto podemos encontrar acciones como: Recabar información, inscribirse para el examen, contestar el examen escrito, asistir a la entrevista personal, matricularse. Todas estas acciones forman parte de un Proceso, todas están organizadas en el tiempo, no es posible dar el examen escrito antes de inscribirse.

De igual forma podríamos mencionar el Proceso de la digestión: Masticar los alimentos, deglutir, alimentos digeridos por los jugos gástricos en el estómago, alimentos que transitan a lo largo del intestino para que el cuerpo obtenga la energía para vivir. Estas acciones no pueden alterar el orden, están organizadas en el tiempo y forman parte de un acontecimiento más grande llamado Digestión.

También podríamos mencionar el proceso de fabricación de zapatos: Comprar el cuero, dar color al cuero, comprar la suela, cortar el cuero, cortar la suela, coser el cuero a la suela, colocar los pasadores, empaquetar, enviar a la tienda. También son procesos organizados en el tiempo, cuyo orden no se puede alterar.

## 1.2 Algoritmo

Así como ocurren muchos procesos en todas las actividades que desarrolla el ser humano, también ocurren procesos en el computador. En este caso les llamamos ALGORITMOS. Pueden ejercitarse definiciones parecidas a las anteriores pero un poco más precisas.

- Una serie de operaciones detalladas y no ambiguas que se ejecutan paso a paso y que conducen a la resolución de un problema
- Un conjunto de reglas para resolver una cierta clase de problema
- Una forma de describir la solución de un problema

Un algoritmo es también un conjunto de acciones organizadas en el tiempo, cuyo orden no puede sufrir alteraciones. También es importante recalcar que las acciones ejecutadas en un algoritmo se desarrollan una a la vez, nunca simultáneamente.

Cada una de las acciones que componen un algoritmo recibe el nombre de SENTENCIA. Cada sentencia debe estar en un lenguaje comprensible para el ejecutor. Depende del momento para definir quien o que es el ejecutor de un algoritmo. Cuando se está diseñando el algoritmo el ejecutor es un usuario, por lo tanto debe estar en un lenguaje comprensible para el ser humano, pues será éste el ejecutor, sin embargo, el algoritmo en cierto momento estará listo para trabajar en un computador en cuyo caso será el computador el ejecutor.

El estudiante de Técnicas de Programación debe tener claro el uso, tanto de lenguajes comprensibles para el usuario como para el computador.

### 1.2.1 Elementos de los Algoritmo

Los algoritmos están compuestos de dos elementos:

- Descripción de acciones a ser ejecutadas. Que de otra forma son las sentencias antes mencionadas. Estas acciones tienen varias formas de plantearse, dependiendo quien es el ejecutor del algoritmo.
- Descripción de los datos que son manipulados por estas acciones, los cuales deben ser declarados o definidos. También son diversas las formas de declarar datos, dependiendo del ejecutor.

Más adelante se estudiarán las formas de plantear acciones y declaraciones en lenguajes para el usuario y para el computador.

### 1.2.2 Características de los Algoritmos

Los algoritmos tienen ciertas características. Es importante mencionar las siguientes:

- Debe ser preciso al indicar el orden de realización de cada paso. Como ya se ha dicho, las acciones se ejecutan una a una, nunca simultáneamente, en un determinado orden. Este orden debe estar definido en forma precisa, no habrán dudas al respecto.
- Debe estar definido. Si se sigue el algoritmo dos o más veces, con los mismos datos, se debe obtener el mismo resultado. Nunca un algoritmo puede proporcionar resultados diferentes basados en circunstancias que no sean los datos y el orden de las acciones.
- Debe ser finito. Si se sigue un algoritmo debe terminar en algún momento. Nunca un algoritmo puede ejecutarse indefinidamente.

## 1.3 Datos

Los datos con los cuales trabaja un algoritmo son la materia prima para obtener información. Los datos con los cuales trabaja un programa se sitúan en las VARIABLES. Una variable es un pequeño espacio en la memoria de computador, al cual se le da un nombre.

Para dar un nombre a una variable debe ser DECLARADA.

### 1.3.1 Tipos de Datos

Básicamente existen tres tipos de datos. Decimos básicamente pues cuando se trabaja con lenguajes de programación se descubren muchos otros tipos de datos, la mayoría de los cuales son sub tipos de los que se van a mencionar. Ahora es importante mencionar estos tres:

- Numéricos: Son datos que solo podrán ser números en cualquiera de las formas que conocemos para los mismos. Por ejemplo: 2, 45.6, -7.
- Alfanuméricos: Son datos que podrán ser números, letras o caracteres especiales. Por ejemplo: AF, xg, 34gt, \$tg, #tf5.
- Lógicos: Son datos que solo pueden contener dos posibles valores: verdadero o falso

En otras palabras, los datos numéricos solo contienen números mientras que los alfanuméricos contienen cualquier carácter. De esto se desprende que los datos con los cuales se pueden hacer operaciones aritméticas son los datos numéricos.

## 1.4 Variables

Como se ha mencionado en el punto 1.3, los datos están contenidos en las variables, que son espacios de memoria en el computador. Las variables tienen dos atributos o características y un valor.

Las características de una variable son: un nombre y un tipo.

El nombre de una variable puede ser cualquier conjunto de caracteres que no empiece con un número. En la siguiente tabla vemos posibles ejemplos de nombres de variables:

X	Es un nombre válido
L3	Es un nombre válido
IMPORTE	Es un nombre válido, además expresa mejor la idea sobre su contenido que los dos nombres anteriores.
IMPORTE_DE_FACTURA	Es un nombre válido. Tiene varias palabras unidas por sub guiones
IMPORTE_FACTURA	Es un nombre válido
IMP FACTURA	NO es válido. Un nombre no puede tener espacios intermedios.
3TIPO	NO es un nombre válido, no puede comenzar con un carácter numérico.

El tipo de una variable coincide con los tipos de datos antes mencionados, es decir una variable puede ser numérica, alfanumérica o lógica.

Los valores contenidos en una variable pueden ir cambiando durante la ejecución del algoritmo, por eso el nombre VARIABLE, es decir implica cambio, modificación. Los contenidos de las variables no cambian libremente por decisión del ejecutor, cambian porque alguna acción así lo decide. Más adelante, en el punto 1.6 se explica como las Expresiones cambian los valores de las variables.

## 1.5 Constantes

Una constante es un objeto usado dentro de un algoritmo cuyo valor es invariable, es decir no cambia durante la ejecución. Una constante no requiere ser declarada, simplemente es usada en el algoritmo. Las constantes pueden ser numéricas o alfanuméricas. Lo que debe tenerse en cuenta es que las constantes alfanuméricas podrían confundirse con los nombres de variables, por lo tanto cuando se usan deben estar encerradas dentro de comillas.

La siguiente tabla muestra posibles constantes y variables.

5	Es una constante numérica
7.89	Es una constante numérica
BLANCO	Es un nombre de variable que contiene algún valor que puede cambiar durante la ejecución del algoritmo.
“BLANCO”	Es una constante alfanumérica cuyo valor son los caracteres “BLANCO”, no cambia durante la ejecución del algoritmo.
-4	Es una constante numérica
“SARA”	Es una constante alfanumérica

## 1.6 Expresiones

Son combinaciones de variables, constantes, operadores aritméticos y operadores de comparación. Se pueden diferenciar dos tipos de expresiones: las expresiones aritméticas y las expresiones booleanas.

Las primeras cambian valores en alguna variable, es decir son acciones propiamente dichas. Las segundas son preguntas o interrogaciones sobre los valores que se encuentran en las variables. Estas preguntas solo pueden tener dos respuestas SI o NO.

### 1.6.1 Expresiones Aritméticas

Las expresiones aritméticas se plantean en forma de ecuación matemática por igualdad, donde a la izquierda del signo = habrá siempre un nombre de variable y a la derecha del signo = cualquier operación matemáticamente válida.

Las operaciones aritméticas son:

+	Suma
-	Resta
*	Multiplicación
/	División

Es posible usar paréntesis cuando es necesario jerarquizar operaciones, es decir, definir cual debe ejecutarse primero dentro de la expresión. La ejecución es de adentro hacia fuera.

Para dar ejemplos de expresiones aritméticas vamos a usar las variables numéricas A, B, C y la variable alfanumérica D, las cuales tienen inicialmente ciertos valores. Cada vez que se ejecuta una expresión aritmética cambia el valor de la variable. (Vaya verificando los cambios en el diagrama de variables).

Los valores iniciales son:

A	0	B	0	C	0	D	
---	---	---	---	---	---	---	--

La expresión  $A = 5$  Coloca la constante 5 en la variable A.

A	5	B	0	C	0	D	
---	---	---	---	---	---	---	--

La expresión  $C = A$  Coloca en C el valor que está en A

A	5	B	0	C	5	D	
---	---	---	---	---	---	---	--

La expresión  $B = A + 3$  Coloca en B la suma de la constante 3 y el valor de A.

A	5	B	8	C	5	D	
---	---	---	---	---	---	---	--

La expresión  $C = C + 1$  Coloca en C la suma del valor que tenía antes de la ejecución más la constante 1. En este caso, a diferencia del anterior, la variable receptora participa también en la operación.

A	5	B	8	C	6	D	
---	---	---	---	---	---	---	--

Si se vuelve a ejecutar la expresión  $C = C + 1$ .

A	5	B	8	C	7	D	
---	---	---	---	---	---	---	--

La expresión  $D = A$

Coloca en la variable D el valor que se encuentra en la variable A. En este caso A es una variable.

A	5	B	8	C	7	D	5
---	---	---	---	---	---	---	---

La expresión  $D = "A"$

Coloca en la variable D el valor A. Es diferente al caso anterior. Ahora el nuevo valor de D se origina en una constante, no en una variable.

A	5	B	8	C	7	D	A
---	---	---	---	---	---	---	---

La expresión  $A = (B + 5) * C$

Coloca en A el valor producido por la suma del valor de B más 5 y luego multiplicado por el valor de C.

A	91	B	8	C	7	D	A
---	----	---	---	---	---	---	---

La expresión  $A = B + (5 * C)$

Coloca en A el valor producido por la multiplicación de valor de C por la constante 5 y luego sumado al valor de B. Note que son las mismas variables y las mismas operaciones, pero los paréntesis han jerarquizado las operaciones de diferente forma. Los resultados son distintos

A	43	B	8	C	7	D	A
---	----	---	---	---	---	---	---

A continuación se plantean algunos casos de expresiones que no son válidas:

$D = "RR" + 5$

No se puede sumar una constante numérica a una constante alfanumérica

$A = "GG"$

Una variable numérica no puede recibir un valor alfanumérico.

$A + 3 = C * 4$

A la izquierda del signo  $=$  solo pueden haber nombres de variables, no operaciones.

$C > A + 4$

Las operaciones aritméticas solo pueden definirse con el signo  $=$ .

## 1.6.2 Expresiones booleanas

Las expresiones booleanas no modifican ningún valor. Plantean preguntas sobre los valores contenidos en las variables. Las preguntas están hechas de forma que solo tienen dos respuestas posibles: SI o NO.

En una expresión booleana se pueden comparar dos variables o se puede comparar una variable con una constante. Las expresiones booleanas pueden hacer uso de diferentes comparadores:

=	igual que
>	mayor que
<	menor que
<>	diferente que
>=	mayor o igual que
<=	menor o igual que

A	5	B	8	C	8	D	FFF
---	---	---	---	---	---	---	-----

Usando los valores de las variables antes indicadas se pueden plantear ejemplos:

Expresión	Interpretación en español	Resultado
A = 6	El valor de la variable A es igual a la constante 6?	NO
A = 5	El valor de la variable A es igual a la constante 5?	SI
B = C	El valor de la variable B es igual al valor de la variable C?	SI
B > C	El valor de la variable B es mayor al valor de la variable C?	NO
B <= C	El valor de la variable B es menor o igual al valor de la variable C?	SI
D = "FF"	El valor de la variable D es igual a la constante "FF"	NO
D = "FFF"	El valor de la variable D es igual a la constante "FFF"	SI
A <> C	El valor de la variable A es diferente al valor de la variable C	SI
D = "A"	El valor de la variable D es igual a la constante "A"	NO

Se pueden plantear dos preguntas combinadas usando los operadores AND y OR. En español AND significa Y, OR significa O.

Cuando se usa AND, para que la respuesta total sea positiva ambas respuestas deben ser positivas.

Cuando se usa OR solo una de las respuestas debe ser positiva para que la respuesta total sea positiva.

En los siguientes cuadros podemos ver la diferencia.

Operador AND		
Preg 1	Preg 2	Total
SI	SI	SI
SI	NO	NO
NO	SI	NO
NO	NO	NO

Operador OR		
Preg 1	Preg 2	Total
SI	SI	SI
SI	NO	SI
NO	SI	SI
NO	NO	NO

Con las siguientes variables y valores se pueden plantear ejemplos de preguntas combinadas:

A	5	B	8	C	8	D	FFF
---	---	---	---	---	---	---	-----

Pregunta Combinada	Respuesta Preg 1	Respuesta Preg. 2	Respuesta Total
A = B OR C > 5	NO	SI	SI
A = B AND C > 5	NO	SI	NO
B = C OR D = "FFF"	SI	SI	SI
B = C AND D = "FFF"	SI	SI	SI
C = 7 OR B < 5	NO	NO	NO
A <> 5 AND B >= 9	NO	NO	NO