

## LENGUAJES

Lenguaje es el empleo de notaciones, señales y vocales (voz, palabras) para expresar ideas, comunicarse, y establecer relaciones entre los seres humanos. Un lenguaje no sólo consta de "palabras", sino también de su pronunciación y los métodos para combinar las palabras en frases y oraciones; los lenguajes se forman mediante combinaciones de palabras definidas en un diccionario terminológico previamente establecido. Las combinaciones posibles deben respetar un conjunto de reglas sintácticas establecidas, a ello se le conoce con el nombre de **Sintaxis**. Además, las palabras deben tener determinado sentido, deben ser comprendidas por un grupo humano en un contexto dado, a ello se le denomina **Semántica**.

## TIPOS DE LENGUAJES

Aunque existen muchas clasificaciones, en general se puede distinguir entre dos clases de lenguajes: los **lenguajes naturales** (inglés, alemán, español, etc.) y los **lenguajes artificiales** o formales (matemático, lógico, computacional, etc.). Tanto el lenguaje natural como el lenguaje artificial son humanos. El primero es natural porque se aprende (o adquiere) inconscientemente e involuntariamente. Ningún bebé decide aprender o no la lengua que hablan sus padres, y ningún padre sienta a su hijo y le enseña las reglas sintácticas de su lengua. Las personas hablan y se entienden, pero generalmente no se cuestionan las reglas que utilizan al hablar. Por otra parte, los lenguajes artificiales sí se aprenden de manera voluntaria y conscientemente. Un ejemplo de lenguaje artificial son los **lenguajes de programación** utilizados para desarrollar programas informáticos.

## LOS LENGUAJES DE PROGRAMACIÓN

Un **Lenguaje de Programación** es un conjunto de reglas, notaciones, símbolos y/o caracteres que permiten a un programador poder expresar el procesamiento de datos y sus estructuras en la computadora. Cada lenguaje posee sus propias sintaxis. También se puede decir que un programa es un conjunto de órdenes o instrucciones que resuelven un problema específico basado en un Lenguaje de Programación.

Existen varias clasificaciones para los lenguajes de programación.

### Clasificación de los Lenguajes de Programación

Los programadores escriben instrucciones en diversos lenguajes de programación. La computadora puede entender directamente algunos de ellos, pero otros requieren pasos de traducción intermedios. Hoy día se utilizan cientos de lenguajes de computadora.

Los Lenguajes de Programación pueden clasificarse **de acuerdo con su uso** en:

1. Lenguajes desarrollados para el **cálculo numérico**. Tales como **FORTRAN**, Mathematica y Matlab.
2. Lenguajes para **sistemas**. Como **C**, **C++** y ensamblador.
3. Lenguajes para aplicaciones de **Inteligencia Artificial**. Tales como Prolog, y Lisp.

También se pueden clasificar de acuerdo con el **tipo de instrucciones de que constan**. En esta clasificación se tiene al **lenguaje máquina**, al **lenguaje ensamblador** y al **lenguaje de alto nivel**. Se presenta a continuación una descripción de cada uno de ellos.

### Lenguaje máquina (Binario)

Una computadora sólo puede entender el lenguaje máquina. El lenguaje de máquina ordena a la computadora realizar sus operaciones fundamentales una por una. Dicho lenguaje es difícil de usar para la persona porque trabajar con números no es muy cómodo además de que estos números están en formato binario. ¿Cómo es que se representan las operaciones como números? John Von Neumann desarrolló el modelo que lleva su nombre para esta representación. Ya se estudió que representar números usando el sistema binario no es complicado, pero se tenía luego el problema de representar las acciones (o instrucciones) que iba a realizar la computadora también en el sistema binario; pues la memoria, al estar compuesta por bits, solamente permite almacenar números binarios. La solución que se tomó fue la siguiente: a cada acción que sea capaz de realizar la computadora, se le asocia un número, que corresponde a su código de operación (**opcode**). Por ejemplo, una calculadora programable simple podría asignar los siguientes opcodes :

1 = SUMA, 2 = RESTA, 3 = MULTIPLICA, 4 = DIVIDE

Supóngase entonces que se quiere realizar la operación  $5 * 3 + 2$ , en la calculadora descrita arriba. En la memoria de la calculadora se podría representar el programa de la siguiente forma:

Posición	Opcode	Significado	Comentario
0	5	5	Primer número de la fórmula
1	3	*	3 es el <b>opcode</b> que representa la multiplicación.
2	3	3	Segundo número de la fórmula
3	1	+	1 es el <b>opcode</b> para la suma.
4	2	2	Último número de la fórmula

y en código binario:

5 3 3 1 2

**101 011 011 001 010**

Podemos ver que, con esta representación, es simple expresar las operaciones de las que es capaz de realizar el hardware en la memoria. La descripción y uso de los opcodes es lo que se llama lenguaje de máquina. El lenguaje máquina es el lenguaje más primitivo y depende directamente del hardware.

### **Lenguajes de bajo nivel** (ensamblador)

Para facilitar y agilizar su labor a los programadores, se buscaron nuevos lenguajes. Cuando abstraemos los opcodes y los sustituimos por una palabra que sea una clave de su significado, se tiene el concepto de Lenguaje Ensamblador. Así, el lenguaje ensamblador representa las acciones del ordenador mediante pequeñas abreviaturas de palabras en inglés. Podemos entonces definir al Lenguaje Ensamblador de la siguiente forma:

**Lenguaje Ensamblador** consiste en asociar a los opcodes palabras clave que faciliten su uso por parte del programador

No obstante, el lenguaje ensamblador requiere de muchas instrucciones para realizar simples operaciones.

### **Lenguajes de alto nivel**

Para acelerar aun más el proceso de programación se desarrollaron los lenguajes de alto nivel, en los que se puede escribir un sólo enunciado para realizar tareas sustanciales. Los lenguajes de alto nivel permiten a los programadores escribir instrucciones que asemejan al inglés cotidiano y contiene notaciones matemáticas de uso común. El concepto de lenguaje de alto nivel nació con el lenguaje FORTRAN (FORmula TRANslation) que, como su nombre indica, surgió como un intento de traducir fórmulas matemáticas al lenguaje ensamblador y por consiguiente al lenguaje de máquina. A partir de FORTRAN, se han desarrollado innumerables lenguajes que siguen el mismo concepto: buscar la mayor abstracción posible y facilitar la vida al programador, aumentando la productividad. Entre estos lenguajes de alto nivel se encuentra el **lenguaje C++** que servirá de base para el desarrollo del curso.

## EJEMPLO DE TIPOS DE LENGUAJES

### Lenguaje Máquina

100001010101010

100100101010100

100011100101110

### Lenguaje de Nivel Bajo (Ensamblador)

LOAD R1, (B)

LOAD R2, (C)

ADD R1, R2

STORE (A), R1

### Lenguajes de Alto Nivel

A = B + C;

---

## HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

Se presentan a continuación datos relevantes de algunos de los lenguajes de programación de mayor importancia histórica.

### **FORTRAN**

FORTRAN es el más viejo de los lenguajes de alto nivel. Fue diseñado por IBM en 1950. El idioma se hizo tan popular en los 60's que otros desarrolladores empezaron a producir sus propias versiones y esto llevó a una gran cantidad de dialectos (en 1963 había 40 compiladores de FORTRAN diferentes). En 1972 se creó *FORTRAN66*, como una forma de estandarizar la estructura del lenguaje. Luego, en 1980, se estableció una norma oficial para el lenguaje avalada por la Organización de Normas Internacionales (ISO). Tal versión es normalmente conocida como *FORTRAN 77* (dado que el proyecto final se completó en 1977). En 1991 surge *FORTRAN90*, un desarrollo mayor del idioma pero que incluye todos los elementos de FORTRAN77 para facilitar la compatibilidad. Finalmente, en 1997, surge *FORTRAN95* o High Performance Fortran (HPF).

### **BASIC**

BASIC es la abreviación de *Beginners All-purpose Symbolic Instruction Code*. Basic fue desarrollado en la Universidad de Dartmouth en 1964 bajo la dirección de J. Kemeny y T. Kurtz. Surgió como un idioma simple de aprender y fácil de traducir. En los 70's, cuando se creó la computadora personal Altair, Bill Gates y Paul Allen implementaron su propia versión de Basic en dicha computadora. Con ello comenzó el futuro de BASIC y de la PC. En ese tiempo, Gates era estudiante de Harvard y Allen era un empleado de Honeywell. La versión BASIC de Gates ocupaba un total de 4KB de memoria incluyendo el código y los datos que se usaron para el código fuente. Luego Gates implementó BASIC en otras plataformas (Apple, Comodor y Atari) y fue a partir de entonces que la corporación de Microsoft empezó su reinado en el mundo de las PC. Más tarde en los 70's, surgió el sistema operativo MS-DOS de Bill Gates que incluía un intérprete de BASIC. La versión distribuida con MS-DOS era GW-BASIC y podía ser ejecutada en cualquier máquina que pudiera ejecutar DOS.

## C

El lenguaje C reúne características de programación tanto de los lenguajes ensambladores como de los lenguajes de alto nivel; este lenguaje posee gran poderío basado en sus operaciones a nivel de bits (propias de **ensambladores**) y la mayoría de los elementos de la programación estructurada de los lenguajes de **alto nivel**. Por ello es que C ha sido el lenguaje preferido para el desarrollo de software de sistemas y aplicaciones profesionales de la programación de computadoras.

En 1970 Ken Thompson de los laboratorios Bell creó la primera versión del lenguaje, la cual podía ejecutarse en el sistema operativo UNIX; a este lenguaje se le llamó lenguaje B y tenía la desventaja de ser lento. En 1971 Dennis Ritchie, con base en el lenguaje B, desarrolló NB que luego cambió su nombre por C. Su diseño incluyó una sintaxis simplificada, la aritmética de direcciones de memoria (permite al programador manipular bits, bytes y direcciones de memoria) y el concepto de apuntador. Además, al ser diseñado para mejorar software de sistemas, se buscó que generase códigos eficientes y una portabilidad total, es decir el que pudiese correr en cualquier máquina. Logrados los objetivos anteriores, C se convirtió en el lenguaje preferido de los programadores profesionales.

## C++

En 1980 Bjarne Stroustrup, también de los laboratorios Bell, adicionó al lenguaje C las características de la *programación orientada a objetos* (incluyendo la ventaja de una biblioteca de funciones orientada a objetos) y lo denominó C con clases. Para 1983 dicha denominación cambió a la de C++.