Francisco Arjonilla Bermúdez

IES Las Fuentezuelas - Desarrollo de Aplicaciones Multiplataforma

Programación

LENGUAJES DE PROGRAMACIÓN

En la actualidad, podemos encontrar un gran número de lenguajes de programación disponibles, la lista completa es prácticamente inacabable y lo cierto es que actualmente unos pocos lenguajes dominan a nivel internacional.

Podemos empezar diferenciándolos por su nivel de abstracción a la hora de introducir algoritmos: el lenguaje máquina, el lenguaje ensamblador, y el lenguaje de alto nivel.

El lenguaje maquina es aquel cuyas instrucciones son directamente entendibles por la computadora y no necesitan traducción posterior para ejecutar el programa, y sus instrucciones se expresan en términos de la unidad de memoria más pequeña (bit) = digito binario 0 o 1.

Un ejemplo de instrucciones de tipo maquina:

0010 0000 1001 1001 1101 1110.

Inmediatamente después se encuentra el lenguaje ensamblador, el cual ya si que requiere de un programa ensamblador para su uso. El lenguaje ensamblador expresa las instrucciones de una forma más natural al usuario a la vez que muy cercana al microcontrolador, ya que cada una de esas instrucciones se corresponde con otra en código máquina.

Algunos ejemplos de ensambladores que continúan siendo relevantes a dia de hoy son:

**WinAsm**

**Easy Code**

**RadASM**

**Fresh IDE**

**Assembler IDE**

Un ejemplo de instrucciones de Ensamblador 8085:

org 100h

comienzo:

mov si, x

mov al, msg2[si]

cmp msg[si], al jne fin:

cmp msg[si], "$" jz final:

inc x

loop comienzo

final:

mov dx, offset msg3

mov ah, 9

int 21h

fin:

ret

msg db "hello world $"

msg2 db "hello world $"

msg3 db "Son iguales $"

x dw 0

Finalmente, entre los lenguajes de alto nivel encontramos a los lenguajes compilados e interpretados. De forma general, la diferencia entre los lenguajes compilados e interpretados es que los primeros usan un compilador para poder traducirlo y ejecutar el programa, mientras que los segundos requieren de un intérprete que traduzca el código al momento de la ejecución.

Algunos ejemplos de compiladores más usados hoy en día:

**Python**

**Java**

**JavaScript**

**C#**

**PHP**

**C/C++**

**R**

**Objective-C**

Algunos ejemplos de interpretadores más conocidos a día de hoy:

**GNU Octave**

**IDL (Interactive Data Language)**

**TK Solver**

**Mathematica**

**MATLAB.**

Las maquinas virtuales sin embargo, no son más que un software capaz de cargar en su interior otro sistema operativo haciéndole creer que es un ordenador de verdad. Tal y como su nombre indica, el concepto es tan sencillo como crear una máquina (PC, consola, móvil o lo que sea) que en vez de ser física es virtual o emulada.

Agunos ejemplos de Maquinas Virtuales que cabe destacar:

**VirtualBox**

**Parallels**

**VMware**

**QEMU**

**Windows Virtual PC**

La programación ha vivido numerosos cambios que continúan evolucionando a día de hoy, que pueden resumirse en la siguiente tabla

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Programación no estructurada | Programación estructurada | Programación Modular | Programación que usa objetos | Programación Orientada a Objetos |
| Flujo de programas de difícil seguimiento | Ordenación del flujo de programas | Organización del código en módulos, funciones y procedimientos | Existencia de objetos con propiedades y métodos disponibles para el programador | Los programas se conciben como un conjunto de objetos representación del problema real |
| Años 60 | Años 70 | Años 80 | Años 90 | Años 2000 |