



UT1. INTRODUCCIÓN A LA PROGRAMACIÓN

Módulo: PROGRAMACIÓN

Curso 2022/2023. 1º DAM

Ruth Lospitao Ruiz

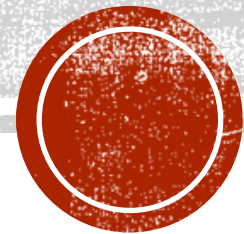


CONTENIDOS

- Conceptos generales
- Lenguajes de programación
- Paradigmas de programación
- Fases de programación
- Ciclo de vida del sw
- Herramientas y entornos para el desarrollo de programas



LENGUAJES DE PROGRAMACIÓN



CONCEPTO

- Un **lenguaje de programación** es un lenguaje formal que proporciona una serie de instrucciones que permiten a un programador escribir secuencias de órdenes y algoritmos a modo de controlar el comportamiento físico y lógico de una computadora con el objetivo de resolver un problema
- A todo este conjunto de órdenes y datos escritos mediante un lenguaje de programación se le conoce como **programa**.



HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

- Cualquier programa ha de escribirse en un lenguaje entendible por el ordenador
- El **ordenador**:
 - Máquina electrónica (no habla nuestro idioma).
 - Dotada principalmente de memoria y un procesador, que es el encargado de interpretar el algoritmo, ejecutando las operaciones correspondientes a cada paso.
 - El proceso de elaboración de un programa o software se denomina **programación**.
 - El conjunto de instrucciones que se pueden utilizar para construir un programa se denomina **lenguaje de programación**.
- Representación de los datos
 - Se realiza de forma binaria (BIT: 0 ó 1).
 - Esta representación está basada en la capacidad que tiene el ordenador de diferenciar dos estados o niveles de voltaje.

HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

- La representación se puede hacer en varios niveles:
 1. **Lenguaje máquina** (código binario: por ejemplo, el conjunto de instrucciones para los microprocesadores de la familia x86).

[illegible]

HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

2. Lenguaje ensamblador (por ejemplo, la sintaxis Intel para el conjunto de instrucciones x86).

```
1. .model small
2. .stack
3. .data
4.     saludo    db "Hola mundo!!!", "$"
5.
6. .code
7.
8. main proc           ;Inicia proceso
9.     mov     ax,seg saludo    ;hmm ¿seg?
10.    mov     ds,ax           ;ds = ax = saludo
11.
12.    mov     ah,09           ;Function(print string)
13.    lea     dx,saludo        ;DX = String terminated by "$"
14.    int     21h             ;Interruptions DOS Functions
15.
16. ;mensaje en pantalla
17.
18.    mov     ax,4c00h         ;Function (Quit with exit code (EXIT))
19.    int     21h             ;Interruption DOS Functions
20.
21. main endp           ;Termina proceso
22. end main
```

Ejemplo de HolaMundo en ensamblador



3. Lenguajes de alto nivel: se caracteriza por expresar los algoritmos de una manera adecuada a la capacidad cognitiva humana, en lugar de la capacidad con que los ejecutan las máquinas.

- máxima flexibilidad al programador a la hora de abstraerse
- permiten un camino bidireccional entre el lenguaje máquina y una expresión casi oral entre la escritura del programa y su posterior compilación.
- Por lo general suelen estar orientados a objetos, a eventos o a funciones, pudiendo estos combinarse.
- Asimismo, pueden ser compilados o interpretados.
- Algunos ejemplos son: Java, PHP, Python, Javascript, C++.



HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

▪ **Lenguajes máquina:**

- El único que entiende directamente el ordenador.
- Están compuestos por cadenas binarias (cadenas de 0s y 1s) que especifican una serie de operaciones y las posiciones de memoria implicadas en la operación.
- El código máquina es conocido como código binario, aunque se emplea el sistema hexadecimal para simplificar el trabajo de escritura.

▪ **Inconvenientes**

- Dificultad y lentitud en la codificación.
 - Poca fiabilidad.
 - Gran dificultad para verificar y poner a punto los programas.
 - Los programas solo son ejecutables en el mismo procesador.
- La única **ventaja** de programar en lenguaje máquina
- Posibilidad de cargar un programa en memoria sin necesidad de traducción posterior ya que el código es perfectamente inteligible por el computador.
- Escribir un programa utilizando cadenas de 0 y 1 implicaría un tiempo elevado y una gran probabilidad de cometer errores debido a la dificultad y lentitud en la codificación. Además los programas sólo son ejecutables en el mismo procesador para el que se ha realizado la codificación. Por esta razón, se desarrollaron los **lenguajes de ensamblador**.

HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

▪ *Lenguajes ensambladores o de bajo nivel*

- Indican al ordenador las operaciones que ha de realizar mediante la utilización de códigos nemotécnicos.
- Son más fáciles de utilizar que los lenguajes máquina, pero, al igual que ellos, dependen de la máquina en cuestión.
- Un programa realizado en lenguaje ensamblador no puede ser ejecutado directamente por la computadora, sino que necesita ser traducido a lenguaje máquina (código binario).
- El programa utilizado para realizar esta conversión se denomina programa ensamblador y se encarga de obtener el programa objeto en código máquina a partir del programa fuente en ensamblador.
- Inconvenientes: Dependencia total de la máquina, lo que impide la portabilidad de la aplicación entre máquinas con procesadores diferentes. Su utilización requiere conocer bien el hardware.
- Ventajas :
 - Respecto al código máquina: mayor facilidad de codificación/modificación, ahorran tiempo y requieren menos atención a detalles.
 - Respecto a lenguajes de alto nivel: velocidad de ejecución.
- Aplicaciones reducidas: Tiempo real, control de procesos y de dispositivos electrónicos.



HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

- ***Lenguaje de alto nivel***
 - Hoy en día son los más utilizados. Están orientados al problema.
 - Permiten una mejor comprensión de los programas debido al empleo de una terminología que se aproxima al lenguaje humano.
 - Los programas son independientes de la máquina, lo que nos permitirá ejecutarlos en diferentes máquinas sin ninguna o pocas modificaciones.
 - El ordenador no es capaz de reconocer directamente estas ordenes, por lo que es necesaria su traducción a un lenguaje que el sistema pueda entender.

HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN

▪ ***Lenguaje de alto nivel***

▪ ***Ventajas***

- Tiempo de formación de los programadores es relativamente corto comparado con otros lenguajes.
- La escritura de programas se basa en reglas sintácticas similares a los lenguajes humanos.
- Las modificaciones y puestas a punto de los programas son más fáciles.
- Reducción del costo de los programas.
- Transportabilidad.
- Permiten tener una mejor documentación.
- Fáciles de mantener.

▪ ***Desventajas***

- No se aprovechan los recursos internos de la máquina que se explotan mucho mejor en lenguajes máquina y ensambladores.
- Aumento de la ocupación de memoria.
- Tiempo de ejecución de los programas es mucho mayor.

EJEMPLOS LENGUAJES ALTO NIVEL

```
public class HolaMundo {  
    public static void main(String[] args) {  
        System.out.println("Hola Mundo");  
    }  
}
```

Java

```
#include <stdio.h>  
  
int main()  
{  
    printf( "Hola mundo." );  
    return 0;  
}
```

C

```
print("Hola Mundo")
```

Python



LENGUAJES DE PROGRAMACIÓN

- Recapitulemos el concepto
 - ¿Qué es lenguaje? Conjunto de símbolos y palabras (vocabulario o léxico) y conjunto de reglas (sintaxis y semántica) que permiten agrupar los símbolos para formar frases del lenguaje
 - ¿De programación? Que sirve para especificar algoritmos sobre un ordenador

UN LENGUAJE DE PROGRAMACIÓN VIENE DEFINIDO POR UN LÉXICO, UNA SINTÁXIS Y UNA SEMÁNTICA





UT1. INTRODUCCIÓN A LA PROGRAMACIÓN

Módulo: PROGRAMACIÓN

Curso 2022/2023. 1º DAM

Ruth Lospitao Ruiz

