**Tema 13**

* **Periférico:** es la denominación genérica para designar al aparato o dispositivo auxiliar e independiente conectado a la [unidad central de procesamiento](https://es.wikipedia.org/wiki/Unidad_central_de_procesamiento) de una [computadora](https://es.wikipedia.org/wiki/Computadora).
* **Arquitectura Von Neumann:** describe una arquitectura de diseño para las computadoras dividida en una unidad de procesamiento, una unidad de control que contiene los registros de programa y contador, una memoria en la que se almacenan datos y mecanismos de entradas y salidas.
* **Arquitectura Harvard:** describe una arquitectura de diseño para las computadoras con pistas de almacenamiento y de señal físicamente separadas para instrucciones y para datos.
* **Sistema operativo**: conjunto de programas que permite manejar la memoria, disco, medios de almacenamiento de información y los diferentes periféricos o recursos de nuestra computadora, como son el teclado, el mouse, la impresora, la placa de red, entre otros
* **Software**: conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.
* **Hardware**: conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.
* **Rasperry :** ordenador de placa reducida, ordenador de placa única u ordenador de placa simple de bajo costo desarrollado en el Reino Unido por la Raspberry Pi Foundation, con el objetivo de estimular la enseñanza de informática en las escuela.
* **Arduino**: placa de microcontrolador de código abierto basado en el microchip ATmega328P y desarrollado por Arduino.cc.​​ La placa está equipada con conjuntos de pines de E/S digitales y analógicas que pueden conectarse a varias placas de expansión y otros circuitos.

**Tema 14**

* **Compilador:** es una herramienta que crea un programa al analizar, interpretar o compilar de algún tipo de descripción formal de un lenguaje al de la máquina
* **Interprete:**  es un [programa informático](https://es.wikipedia.org/wiki/Programa_(computaci%C3%B3n)) capaz de analizar y ejecutar otros programas. Los intérpretes se diferencian de los [compiladores](https://es.wikipedia.org/wiki/Compilador) o de los [ensambladores](https://es.wikipedia.org/wiki/Ensamblador) en que mientras estos traducen un programa desde su descripción en un [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) al [código de máquina](https://es.wikipedia.org/wiki/C%C3%B3digo_de_m%C3%A1quina) del sistema, los intérpretes sólo realizan la [traducción](https://es.wikipedia.org/wiki/Proceso_de_traducci%C3%B3n_de_programas) a medida que sea necesaria, típicamente, instrucción por instrucción, y normalmente no guardan el resultado de dicha traducción.
* **Lenguaje de programación:** es un [lenguaje formal](https://es.wikipedia.org/wiki/Lenguaje_formal) (o artificial, es decir, un lenguaje con reglas gramaticales bien definidas) que le proporciona a una persona, en este caso el programador, la capacidad de escribir (o programar) una serie de [instrucciones](https://es.wikipedia.org/wiki/Instrucci%C3%B3n_(inform%C3%A1tica)) o secuencias de órdenes en forma de [algoritmos](https://es.wikipedia.org/wiki/Algoritmo) con el fin de controlar el comportamiento físico o lógico de una [computadora](https://es.wikipedia.org/wiki/Computadora), de manera que se puedan obtener diversas clases de datos o ejecutar determinadas tareas
* **Editor de código:** un editor de código fuente es un editor de texto diseñado específicamente para editar el código fuente de programas informáticos. Puede ser una aplicación individual o estar incluido en un entorno de desarrollo integrado
* **Lenguaje de alto nivel:**  es aquel que se aproxima más al lenguaje natural humano que al lenguaje [binario](https://www.ecured.cu/Sistema_binario) de las computadoras, el que se conoce como lenguaje de bajo nivel. Un lenguaje de alto nivel permite al programador escribir las instrucciones de un programa utilizando palabras o expresiones sintácticas muy similares al inglés
* **Lenguaje de máquina:** es el sistema de códigos directamente interpretable por un circuito microprogramable, como el [microprocesador](https://es.wikipedia.org/wiki/Microprocesador) de una [computadora](https://es.wikipedia.org/wiki/Computadora_electr%C3%B3nica) o el [microcontrolador](https://es.wikipedia.org/wiki/Microcontrolador) de un [autómata](https://es.wikipedia.org/wiki/Aut%C3%B3mata_programable). Este lenguaje está compuesto por un [conjunto de instrucciones](https://es.wikipedia.org/wiki/Conjunto_de_instrucciones) que determinan acciones a ser tomadas por la máquin

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Tema 15**

* **Lenguaje de programación:** es un [lenguaje formal](https://es.wikipedia.org/wiki/Lenguaje_formal) (o artificial, es decir, un lenguaje con reglas gramaticales bien definidas) que le proporciona a una persona, en este caso el programador, la capacidad de escribir (o programar) una serie de [instrucciones](https://es.wikipedia.org/wiki/Instrucci%C3%B3n_(inform%C3%A1tica)) o secuencias de órdenes en forma de [algoritmos](https://es.wikipedia.org/wiki/Algoritmo) con el fin de controlar el comportamiento físico o lógico de una [computadora](https://es.wikipedia.org/wiki/Computadora), de manera que se puedan obtener diversas clases de datos o ejecutar determinadas tarea
* **Python:** es un [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) [interpretado](https://es.wikipedia.org/wiki/Int%C3%A9rprete_(inform%C3%A1tica)) cuya filosofía hace hincapié en la legibilidad de su [código](https://es.wikipedia.org/wiki/Codigo_fuente).​ Se trata de un lenguaje de programación [multiparadigma](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n#Paradigma_de_programación), ya que soporta [orientación a objetos](https://es.wikipedia.org/wiki/Programaci%C3%B3n_orientada_a_objetos), [programación imperativa](https://es.wikipedia.org/w/index.php?title=Programaci%C3%B3n_imperativa&action=edit&redlink=1) y, en menor medida, [programación funcional](https://es.wikipedia.org/wiki/Programaci%C3%B3n_funcional). Es un [lenguaje interpretado](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_interpretado), [dinámico](https://es.wikipedia.org/wiki/Tipado_din%C3%A1mico) y [multiplataforma](https://es.wikipedia.org/wiki/Multiplataforma).
* **R:** es un entorno y [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) con un enfoque al [análisis estadístico](https://es.wikipedia.org/wiki/An%C3%A1lisis_estad%C3%ADstico).
* **Lisp:** es una familia de [lenguajes de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) de [computadora](https://es.wikipedia.org/wiki/Computadora_electr%C3%B3nica) de tipo [multiparadigma](https://es.wikipedia.org/wiki/Programaci%C3%B3n_multiparadigma) con larga historia y una inconfundible y útil sintaxis [homoicónica](https://es.wikipedia.org/wiki/Homoiconicidad) basada en la [notación polaca](https://es.wikipedia.org/wiki/Notaci%C3%B3n_polaca).
* **Java:** uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de [cliente-servidor](https://es.wikipedia.org/wiki/Cliente-servidor) de web, con unos diez millones de usuarios reportados.
* **Lenguaje de alto nivel:** lenguaje que se caracteriza por expresar los [algoritmos](https://es.wikipedia.org/wiki/Algoritmo) de una manera adecuada a la capacidad [cognitiva](https://es.wikipedia.org/wiki/Cognici%C3%B3n) humana, en lugar de la capacidad con que los ejecutan las máquinas. Estos lenguajes permiten una máxima flexibilidad al programador a la hora de abstraerse o de ser literal. Permiten un camino bidireccional entre el lenguaje máquina y una expresión casi oral entre la escritura del programa y su posterior compilación
* **Lenguaje de bajo nivel:** lenguaje en el que sus instrucciones ejercen un control directo sobre el hardware y están condicionados por la estructura física de las computadoras que lo soportan
* **Lenguaje compilado:** [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) cuyas [implementaciones](https://es.wikipedia.org/w/index.php?title=Implementaciones&action=edit&redlink=1) son normalmente [compiladores](https://es.wikipedia.org/wiki/Compilador) (traductores que generan [código de máquina](https://es.wikipedia.org/wiki/C%C3%B3digo_de_m%C3%A1quina) a partir del [código fuente](https://es.wikipedia.org/wiki/C%C3%B3digo_fuente)) y no [intérpretes](https://es.wikipedia.org/wiki/Int%C3%A9rprete_(inform%C3%A1tica)) (ejecutores paso a paso del código fuente, donde no se lleva a cabo una traducción en la preejecución).
* **Lenguaje interpretado:** es un [lenguaje informático](https://es.wikipedia.org/wiki/Programa_(computaci%C3%B3n)) capaz de analizar y ejecutar otros programas. Los intérpretes se diferencian de los [compiladores](https://es.wikipedia.org/wiki/Compilador) o de los [ensambladores](https://es.wikipedia.org/wiki/Ensamblador) en que mientras estos traducen un programa desde su descripción en un [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) al [código de máquina](https://es.wikipedia.org/wiki/C%C3%B3digo_de_m%C3%A1quina) del sistema, los intérpretes sólo realizan la [traducción](https://es.wikipedia.org/wiki/Proceso_de_traducci%C3%B3n_de_programas) a medida que sea necesaria, típicamente, instrucción por instrucción, y normalmente no guardan el resultado de dicha traducción.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_-**

* **Algoritmo:** Conjunto ordenado de operaciones sistemáticas que permite hacer una cálculo y hallar la solución de un tipo de problemas.
* **Diagrama de flujo:** Es la representación gráfica de un algoritmo.