# Software

* Todos los programas que pueden ser ejecutados por un sistema de computación. Existe gran variedad para realizar infinitas operaciones.

# Programación

* Proceso de diseñar, codificar, depurar y mantener.

# Algoritmos

* Diseño es la primera fase de un buen software.
* Programa: conjunto de algoritmos que resuelven problemas.
* Algoritmo: conjunto de instrucciones definidas, ordenadas y finitas, para resolver un problema o realizar una actividad.
* El conjunto de instrucciones debe de estar ordenado de forma que no existan dudas sobre que instrucción se va a realizar en cada momento.

# Algoritmia

* Es el estudio de los algoritmos. Se estudian diferentes formas de crearlos.
* Se analiza la efectividad de los algoritmos computacionalmente hablando.
* La inventiva es la principal cualidad para crear buenos algoritmos.
* La forma habitual para representar algoritmos es mediante diagramas de flujo, que representan gráficamente un algoritmo. Facilita la comprensión y el diseño.

# Lenguajes de programación

* Tras diseñarlo, pasamos a codificar. Utilizamos lenguajes de programación, que proporcionan el conjunto de órdenes para componer el algoritmo. Hay muchos lenguajes.

# Tipos de lenguajes

## Lenguajes de bajo nivel

* Instrucciones muy cercanas a las que entiende la máquina.
  + Ventajas: rápidos con control total de los recursos del sistema.
  + Desventajas: muy complicados. Dependen de la arquitectura de la máquina.

## Lenguaje compilado

* El código se compila para crear el ejecutable.
  + Ventajas: complejos, optimizados.
  + Inconvenientes: compilación, dependientes de la plataforma.

## Lenguaje interpretado

* Necesita un intérprete para poder ejecutar el programa creado.
  + Ventajas: rápidos, sencillos, multiplataforma.
  + Inconvenientes: ejecución más lenta y necesitamos un intérprete.

# Historia de la programación

## Java

* Lenguaje más usado en muchos sectores.
* POO.
* Sintaxis derivada de C y C++.
* Se compila a un archivo intermedio: *bytecode*. Es interpretado por una Java Virtual Machine.
* Recolector de basura para liberar memoria.

# Java

* Lenguaje compilado, interpretado, concurrente y orientado a objetos.
* Fuertemente tipado.
* Tiene un intérprete que hace que sea multiplataforma.
* De los más utilizados del mundo.
* Muchas librerías y frameworks.

## Características

* POO, aunque también sirve la programación estructurada.
* Multiplataforma. El mismo código puede utilizarse en varios S.O.
* Pensado para trabajo en red y remoto.
* Optimizado para el uso de la concurrencia.
* Fácil de usar, con sintaxis sencilla y parecida a otros lenguajes como C o C++.
* El programador no hace uso de la memoria del sistema.
  + No usa punteros.
  + Posee un recolector de basura.
* Se pueden crear aplicaciones de distintos tipos:
  + Aplicaciones de escritorio.
  + Sistemas de servidor.
  + Clientes web.
  + Aplicaciones móviles (Android).