

# T1. Introducción a la Informática

## Objetivos

1. Conocer los conceptos básicos y las funciones de un computador.
2. Conocer los elementos necesarios para el tratamiento automático de la información.
3. Conocer y comprender la codificación binaria.
4. Conocer los lenguajes de Programación.

## Contenidos

1. Conceptos Básicos.
2. Codificación de la Información.
3. Componentes Hardware.
4. Lenguajes de Programación.

---

## Conceptos Básicos

### Informática

- Es la abreviatura de **INFORM**ación y auto**MÁTICA**. En inglés se denomina *Computer Science*.
- Estudia el tratamiento automatizado de la información, incluyendo como aspectos más relevantes:
  - El diseño de ordenadores.
  - La programación de ordenadores.
  - El procesamiento de la información, destacando:
    - La resolución de problemas mediante algoritmos.
    - El estudio de los algoritmos en sí mismos.

# Definiciones

## Informática:

- Conjunto de conocimientos y técnicas que permiten recoger, almacenar, organizar, tratar y transmitir datos mediante ordenadores.
- Conjunto de ciencias, técnicas y/o actividades que se dedican al estudio, tratamiento, almacenamiento y transmisión de la información por medios automáticos.

## Información:

- Comunicación o adquisición de conocimientos que permiten ampliar o precisar los que se poseen sobre una materia determinada.

## Dato:

- Representación de una información de manera adecuada para su tratamiento por un ordenador.
- 

# Tipos de información

## Analógica:

La información se representa mediante una magnitud física de tipo continuo (voltaje, tamaño, intensidad, etc.). La cantidad de información almacenada depende de la precisión, pues debemos de ser capaces de diferenciar entre dos valores diferentes.

## Digital:

Se representa mediante algo que puede estar en un número finito de estados. Por ejemplo, para guardar algo en binario hace falta que podamos distinguir entre 2 estados (encendido o apagado, etc.).

# Codificación de la información

La codificación es el proceso de representar símbolos o secuencias de un alfabeto mediante los símbolos o secuencias de otro. Para que la información pueda ser tratada de manera automática mediante ordenadores, ésta debe estar representada de manera adecuada. Se emplea el sistema [binario](#) o en base 2:

## ¿Qué es un bit?

- Un [bit](#) (**BI**nary **di**gi**T**) es la unidad mínima de información.
- Puede representarse de forma numérica mediante los valores 0 y 1.
- Muy adecuado para ordenadores, puesto que la electrónica distingue muy fácilmente entre dos estados: paso/no paso de corriente,...

## Agrupaciones de bits

- Mediante la agrupación de bits es posible codificar más información.
- Con n bits pueden codificarse 2 elevado a n valores diferentes.
- La agrupación de 8 bits se llama [byte](#), permite codificar 256 valores distintos.

---

# Codificación de la información

- A los informáticos les gustan mucho la potencias de dos. La cantidad de información se suele medir en [múltiplos 1024](#) porque 1024 es 2 elevado a 10 y 1000 se parece al "kilo" del sistema internacional de medidas:

Unidades básicas de información (en bytes)				
<a href="#">Prefijos del Sistema Internacional</a>		Prefijo binario		
Múltiplo - (Símbolo)	Estándar <a href="#">SI</a>	Binario	Múltiplo - (Símbolo)	Valor
<a href="#">kilobyte</a> (kB)	10 <sup>3</sup>	2 <sup>10</sup>	<a href="#">kibibyte</a> (KiB)	2 <sup>10</sup>
<a href="#">megabyte</a> (MB)	10 <sup>6</sup>	2 <sup>20</sup>	<a href="#">mebibyte</a> (MiB)	2 <sup>20</sup>
<a href="#">gigabyte</a> (GB)	10 <sup>9</sup>	2 <sup>30</sup>	<a href="#">gibibyte</a> (GiB)	2 <sup>30</sup>
<a href="#">terabyte</a> (TB)	10 <sup>12</sup>	2 <sup>40</sup>	<a href="#">tebibyte</a> (TiB)	2 <sup>40</sup>
<a href="#">petabyte</a> (PB)	10 <sup>15</sup>	2 <sup>50</sup>	<a href="#">pebibyte</a> (PiB)	2 <sup>50</sup>
<a href="#">exabyte</a> (EB)	10 <sup>18</sup>	2 <sup>60</sup>	<a href="#">exbibyte</a> (EiB)	2 <sup>60</sup>
<a href="#">zettabyte</a> (ZB)	10 <sup>21</sup>	2 <sup>70</sup>	<a href="#">zebibyte</a> (ZiB)	2 <sup>70</sup>
<a href="#">yottabyte</a> (YB)	10 <sup>24</sup>	2 <sup>80</sup>	<a href="#">yobibyte</a> (YiB)	2 <sup>80</sup>

Véase tambien: [Nibble](#) · [Byte](#) · [Octal](#)

# Nociones de binario

## Entender estos chistes:

- Hay 10 clases de personas: las que saben binario y las que no.
- Desde que esos informáticos llevan los "100 montaditos" que solo tienen 4 tipos de bocata... ¡y el 1000 hojas sólo tiene 8 capas!
- ¿Cuántos perros tenía la peli de Disney? ...

## Ahora en serio, debéis ser capaces de:

- Convertir un número de binario a decimal y viceversa. Pincha [aquí](#)
- Saber cuántos bits hacen falta como mínimo para representar un rango de valores.
- Qué rango de valores enteros (con y sin signo) se pueden representar con n bits.
- Cuánto ocupa aproximadamente un documento, una foto, canción, película, etc.

---

## Convertir

### De binario a decimal

Sumar potencias de dos. [Enlace wikipedia](#)

### De decimal a binario

Dividir entre 2 de manera sucesiva. [Enlace wikipedia](#)

### Saber número de bits necesarios

La primera potencia de 2 mayor o igual que el número de cosas a discernir

### Cuánto ocupa algo

Razonar sobre el número de bits o de bytes necesarios. Ejemplos:

- ¿cuánto ocupa una imagen de 800x600 pixels en [true color o color verdadero](#)?
- ¿cuánto ocupa un documento de 200 páginas de 25 líneas a 80 columnas?

# Componentes Hardware

*Recuerda que este tema es semipresencial, así que...*

## Debes estudiar:

- El pdf de introducción a la informática que os hemos dejado en poliformaT
  - [La siguiente página de wikipedia](#) donde explica las distintas partes de una computadora:
    - Procesador o Unidad Central de Proceso (CPU)
    - Memoria
    - Dispositivos de entrada/salida
    - Buses o puertos de conexión
  - [Arquitectura de Von Neumann](#)
  - [La jerarquía de memoria](#)
- 

## Lenguajes de programación

- **Algoritmo** es una descripción precisa de cómo resolver un problema
- **Programa** conjunto de instrucciones en una secuencia determinada que nos permiten resolver un problema. Es la implementación de un algoritmo en un lenguaje de programación determinado.
- **Lenguaje de programación** Conjunto de reglas, símbolos y palabras especiales utilizados para construir un programa de forma que pueda ser entendido por el ordenador (quizás tras un proceso de traducción por parte de un [compilador](#) o bien directamente gracias a un [intérprete](#)). Hay un buen [puñado](#)!

## Los lenguajes de programación se clasifican en:

- **Lenguajes Máquina** Es aquél que entiende directamente el ordenador ya que sus instrucciones son secuencias binarias.
- **Lenguajes de Bajo Nivel o Ensambladores.** Compuesto por nemotécnicos de instrucciones de lenguaje máquina. Más fácil de utilizar que el anterior.
- **Lenguajes de Alto Nivel** Son los más utilizados. Mayor facilidad. Programas portables. Requiere traducción del código fuente.

# ¿Por qué vamos a estudiar C?

---

## ¿Por qué C?

- Uno de los más utilizados en ingeniería. Muchos microcontroladores y dispositivos de captura de datos se programan en C.
- Tus sistemas operativos favoritos (linux, mac osx, windows,...) están desarrollados en C.
- Es el núcleo de otros lenguajes como C++ y Objective-C (por ejemplo, aplicaciones mac/iPhone).
- Los conceptos básicos que aprendas de C te servirán para una gran familia de lenguajes como Java, Javascript (aplicaciones web), Matlab (algoritmos numéricos, etc.), etc.
- Aunque no lo creas, te va a servir tanto en la carrera de ingeniería mecánica como en muchos tipos de trabajo de tu campo.
- Aprende a programar te obliga a estructurar el pensamiento y planificar un diseño muy detallado, el mismo tipo de habilidades que necesitas para diseñar otros tipos de sistemas, máquinas, etc.