

2017/2018

UT1: INTRODUCCIÓN A LOS SISTEMAS INFORMÁTICOS.

Sistemas Informáticos

C.F.G.S. Desarrollo de Aplicaciones Multiplataforma



1. INTRODUCCIÓN	3
2. EL SISTEMA INFORMÁTICO, SOFTWARE Y HARDWARE.....	3
2.1. EVOLUCIÓN HISTÓRICA DE LA INFORMÁTICA.	4
2.2. GENERACIONES DE ORDENADORES.	5
3. COMPONENTES HARDWARE.	8
4. COMPONENTES SOFTWARE.	9
5. REPRESENTACIÓN DE LA INFORMACIÓN.....	9
5.1. TIPOS DE DATOS.	10
5.2. SISTEMAS DE NUMERACIÓN.....	11
5.3. CODIFICACIÓN NUMÉRICA.	12
5.4. CAMBIOS DE BASE DE NUMERACIÓN.....	14
5.4.1. CAMBIOS DE BINARIO A DECIMAL.....	14
5.4.2. CAMBIOS DE DECIMAL A BINARIO.....	14
5.4.3. CAMBIOS DE BINARIO A OCTAL	16
5.4.4. CAMBIOS DE OCTAL A BINARIO.	17
5.4.5. CAMBIOS DE OCTAL A DECIMAL.....	17
5.4.6. CAMBIOS DE BINARIO A HEXADECIMAL.....	17
5.4.7. CAMBIOS DE HEXADECIMAL A BINARIO	18
5.4.8. CAMBIOS DE HEXADECIMAL A DECIMAL.....	18
5.5. CODIFICACIÓN ALFANUMÉRICA	18
6. OPERACIONES EN BINARIO.....	20
6.1. SUMA Y RESTA EN BINARIO.....	20
6.2. MULTIPLICACIÓN EN BINARIO.....	22
6.3. DIVISIÓN EN BINARIO	23
7. MEDIDAS DE INFORMACIÓN	23



1. INTRODUCCIÓN.

La informática ha cambiado radicalmente nuestra forma de trabajar y de vivir en general. Desde tiempos remotos el ser humano ha fabricado herramientas que facilitaran sus labores. A medida que se desarrollaba la tecnología, se iban construyendo y perfeccionando máquinas que realizaban cálculos cada vez más complejos, hasta la aparición de los ordenadores. Con ellos se ha llegado a la introducción de la informática en todas las áreas de nuestra vida: el trabajo, el hogar, el ocio, las comunicaciones, la forma de establecer relaciones entre los seres humanos...

En este tema vamos a ver una introducción a los conceptos básicos de la informática, conoceremos los diferentes elementos de los ordenadores y haremos una breve descripción de su funcionamiento tanto interna como externamente.

2. EL SISTEMA INFORMÁTICO, SOFTWARE Y HARDWARE.

Un **sistema informático** es un conjunto de componentes físicos (**hardware**) y lógicos (**software**). El elemento central del sistema informático es el ordenador.

Un ordenador está formado por un conjunto de componentes electrónicos que no pueden realizar ninguna función por sí mismos. Necesitan a su vez de otros componentes no físicos (**software**) que los pongan en funcionamiento. Para que los componentes electrónicos de un ordenador sean capaces de funcionar y realizar un proceso determinado, es necesario que en él se ejecuten un conjunto de órdenes o instrucciones. Estas instrucciones, ordenadas y agrupadas de forma adecuada, constituyen un **programa**.

Pero un programa no funciona por sí solo. Es decir, tenemos los componentes electrónicos; tenemos los programas que incluyen los datos necesarios que se tienen que procesar, pero sigue faltando algo. El componente que falta, que también es un componente software y es el objeto de este módulo, es el **sistema operativo**.

El **sistema operativo** es el componente software de un sistema informático capaz de hacer que los programas (software) procesen información (datos) sobre los componentes electrónicos de un ordenador o sistema informático (hardware).

Físicamente, de mayor a menor precio, tamaño, potencia, velocidad y prestaciones, podemos hacer una clasificación de los sistemas informáticos en:

- ✗ **Superordenadores.** Son ordenadores que no se encuentran en el mercado, sino que se fabrican para grandes organizaciones nacionales o internacionales con el fin de realizar tareas específicas avanzadas como científicas, militares, tecnológicas, ...
- ✗ **Macroordenadores.** (mainframes) Son ordenadores que pueden ocupar grandes espacios y pueden dar servicios a cientos de usuarios simultáneamente.
- ✗ **Servidores y estaciones de trabajo (Workstation).** Son ordenadores que pueden ofrecer servicios a otros ordenadores dentro de una red, como conexión a Internet, acceso a los periféricos, acceso a bases de datos compartidas,... El tamaño de los servidores puede ser más grande o más pequeño, dependiendo de los servicios



ofrecidos y del tamaño de la red. Las estaciones de trabajo, aunque tengan más prestaciones que los ordenadores personales, no suelen ser mucho más grandes que estos.

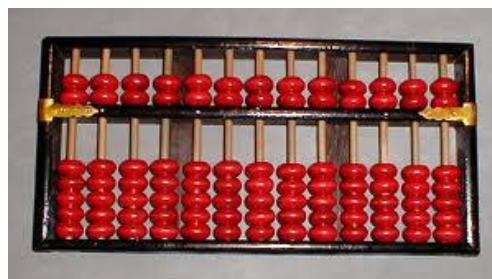
- ✗ **Ordenadores personales o PC.** Son ordenadores que surgieron tras la incorporación de los microprocesadores. Se utilizan tanto en el ámbito doméstico como en el profesional. A su vez pueden ser de sobremesa o portátiles.

2.1. EVOLUCIÓN HISTÓRICA DE LA INFORMÁTICA.

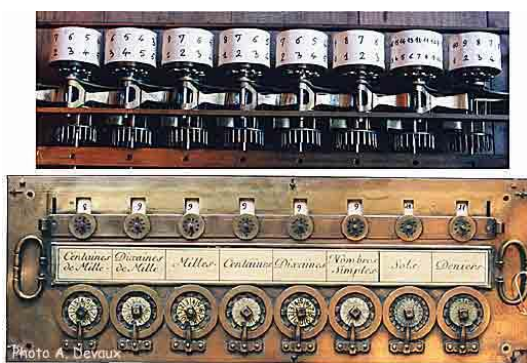
La palabra informática procede de las palabras **información automática**, que quiere decir procesar o automatizar la información.

En este apartado se resume de una forma breve la historia de la informática incluyendo en la última parte las cinco generaciones en las que se divide comúnmente la evolución de los ordenadores.

El dispositivo de cálculo más antiguo que se conoce es el **ábaco**. Se sabe que los griegos empleaban tablas para contar en el siglo V antes de Cristo o tal vez antes. El ábaco tal como lo conocemos actualmente está constituido por una serie de hilos con cuentas ensartadas en ellos. En nuestro país este tipo de ábaco lo hemos visto todos en las salas de billar.



El primer calculador mecánico apareció en 1642. El artífice de esta máquina fue el filósofo francés **Blaise Pascal** (1.623-1.662) en cuyo honor se llama Pascal uno de los lenguajes de programación que más impacto ha causado en los últimos años. A los 18 años Pascal deseaba dar con la forma de reducir el trabajo de cálculo de su padre que era un funcionario de impuestos. La calculadora que inventó Pascal tenía el tamaño de un cartón de tabaco y su principio de funcionamiento era el mismo que rige los cuentakilómetros de los coches actuales; una serie de ruedas tales que cada una de ellas hacía avanzar un paso a la siguiente al completar una vuelta. Las ruedas estaban marcadas con números del 0 al 9 y había dos para los decimales y 6 para los enteros con lo que podía manejar números entre 000.000 01 y 999.999 99.

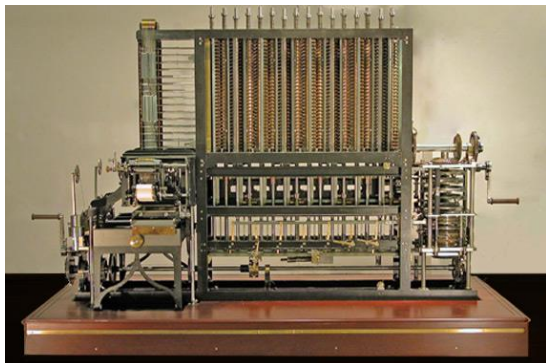


Las ruedas giraban mediante una manivela con lo que para sumar o restar lo que había que hacer era girar la manivela correspondiente en un sentido o en otro el número de pasos adecuado.

Leibnitz (1646-1716) fue uno de los genios de su época; a los 26 años aprendió matemáticas de modo autodidacta y procedió a inventar el cálculo. Inventó una máquina de calcular por la simple razón de que nadie le enseñó las tablas de multiplicar. La máquina de Leibnitz se diferenciaba de la de Pascal en varios aspectos fundamentales el más importante de los cuales era que podía multiplicar dividir y obtener raíces cuadradas.



Aunque hubo muchos precursores de los actuales sistemas informáticos para muchos especialistas la historia empieza con **Charles Babbage**, matemático e inventor inglés que al principio del siglo XIX predijo muchas de las teorías en que se basan los actuales ordenadores. Desgraciadamente al igual que sus predecesores vivieron en una época en que, ni la tecnología, ni las necesidades estaban al nivel de permitir la materialización de sus ideas.



La sofisticada organización de esta máquina, la máquina diferencial según se la llamó es lo que hace que muchos consideren a Babbage padre de la informática actual.

Como los modernos computadores la máquina de Babbage tenía un mecanismo de entrada y salida por tarjetas perforadas, una memoria, una unidad de control y una unidad aritmético-lógica. Preveía tarjetas separadas para programas y datos. Una de sus características más importantes era que la máquina podía alterar su secuencia de operaciones en base al resultado de cálculos anteriores algo fundamental en los ordenadores modernos.

Otro inventor digno de mención es **Herman Hollerith** a los 19 años. El sistema inventado por Hollerith utilizaba tarjetas perforadas en las que mediante agujeros se representaba el sexo, la edad, raza, etc. En la máquina las tarjetas pasaban por un juego de contactos que cerraban un circuito eléctrico activándose un contador y un mecanismo de selección de tarjetas.

Desde 1880 a 1890 la población subió de 50 a 63 millones de habitantes aun así el censo de 1890 se realizó en dos años y medio gracias a la máquina de Hollerith.

Ante las posibilidades comerciales de su máquina Hollerith dejó las oficinas del censo en 1896 para fundar su propia Compañía la Tabulating Machine Company. En 1900 había desarrollado una máquina que podía clasificar 300 tarjetas por minuto una perforadora de tarjetas y una máquina de cómputo semiautomática. En 1924 Hollerith fusionó su compañía con otras dos para formar la Internacional Business Machines, hoy mundialmente conocida como IBM.

2.2. GENERACIONES DE ORDENADORES.

Teniendo en cuenta las diferentes etapas de desarrollo que tuvieron las computadoras, se consideran las siguientes divisiones como generaciones aisladas con características propias de cada una, las cuáles se enuncian a continuación.

Primera Generación (1940 - 1959)

Características Principales:

- Sistemas constituidos por tubos de vacío, desprendían bastante calor y tenían una vida relativamente corta.
- Máquinas grandes y pesadas. Se construye el ordenador ENIAC de grandes dimensiones (30 toneladas).



- Alto consumo de energía. El voltaje de los tubos era de 300v y la posibilidad de fundirse era grande.
- Almacenamiento de la información en tambor magnético interior. Un tambor magnético estaba en el interior del ordenador, recogía y memorizaba los datos y los programas que se le suministraban.
- Continuos fallos o interrupciones en el proceso.
- Requerían sistemas auxiliares de aire acondicionado especial.
- Programación en lenguaje máquina, consistía en largas cadenas de bits, de ceros y unos, por lo que la programación resultaba larga y compleja.
- Alto costo.
- Uso de tarjetas perforadas para suministrar datos y los programas.
- Computadora representativa UNIVAC y utilizada en las elecciones presidenciales de los EE.UU. en 1952.
- Fabricación industrial. La iniciativa se aventuró a entrar en este campo e inició la fabricación de computadoras en serie.

Segunda Generación (1960 - 1965)

Transistores

Cuando los tubos de vacío eran sustituidos por los transistores; estos últimos eran más económicas, más pequeñas que las válvulas miniaturizadas consumían menos y producían menos calor. Por todos estos motivos, la densidad del circuito podía ser aumentada sensiblemente, lo que quería decir que los componentes podían colocarse mucho más cerca unos de otros y ahorrar mucho más espacio.

Características Principales:

- Transistor como componente principal. El componente principal es un pequeño trozo de semiconductor, y se expone en los llamados circuitos transistorizados.
- Disminución del tamaño.
- Disminución del consumo y de la producción del calor.
- Su fiabilidad alcanza metas inimaginables con los efímeros tubos al vacío.
- Mayor rapidez, la velocidad de las operaciones ya no se mide en segundos sino en milésimas de segundos.
- Memoria interna de núcleos de ferrita.
- Instrumentos de almacenamiento: cintas y discos.
- Mejoran los dispositivos de entrada y salida, para la mejor lectura de tarjetas perforadas, se disponía de células fotoeléctricas.
- Introducción de elementos modulares.
- Aumenta la confiabilidad.
- Las impresoras aumentan su capacidad de trabajo.
- Lenguajes de programación más potentes, ensambladores y de alto nivel (Fortran, Cobol y Algol).
- Aplicaciones comerciales en aumento, para la elaboración de nóminas, facturación y contabilidad, etc.

Tercera Generación (1966 - 1971)

Circuito integrado (chips)



Características Principales:

- Circuito integrado desarrollado en 1958 por Jack Kilbry.
- Circuito integrado, miniaturización y reunión de centenares de elementos en una placa de silicio o (chip).
- Menor consumo de energía.
- Apreciable reducción de espacio.
- Aumento de fiabilidad y flexibilidad.
- Aumenta la capacidad de almacenamiento y se reduce el tiempo de respuesta.
- Generalización de lenguajes de programación de alto nivel.
- Compatibilidad para compartir software entre diversos equipos.
- Computadoras en Serie 360 IBM.
- Multiprogramación: Computadora que pueda procesar varios Programas de manera simultánea.
- Tiempo Compartido: Uso de una computadora por varios clientes a tiempo compartido, pues el aparato puede discernir entre diversos procesos que realiza simultáneamente.
- Ampliación de aplicaciones: en Procesos Industriales, en la Educación, en el Hogar, Agricultura, Administración, Juegos, etc.
- La minicomputadora.

Cuarta Generación (1971 - 1981)

Microcircuito integrado

El microprocesador: el proceso de reducción del tamaño de los componentes llega a operar a escalas microscópicas. La microminiaturización permite construir el microprocesador, circuito integrado que rige las funciones fundamentales del ordenador.

Las aplicaciones del microprocesador se han proyectado más allá de la computadora y se encuentra en multitud de aparatos, sean instrumentos médicos, automóviles, juguetes, electrodomésticos, etc.

Memorias Electrónicas: Se desechan las memorias internas de los núcleos magnéticos de ferrita y se introducen memorias electrónicas, que resultan más rápidas. Al principio presentan el inconveniente de su mayor costo, pero este disminuye con la fabricación en serie.

Características Principales:

- Microprocesador: Desarrollado por Intel Corporation a solicitud de una empresa Japonesa (1971).
- El Microprocesador: Circuito Integrado que reúne en la placa de Silicio las principales funciones de la Computadora y que va montado en una estructura que facilita las múltiples conexiones con los restantes elementos.
- Se minimizan los circuitos, aumenta la capacidad de almacenamiento.
- Reducen el tiempo de respuesta.
- Gran expansión del uso de las Computadoras.
- Memorias electrónicas más rápidas.
- Sistemas de tratamiento de bases de datos.
- Generalización de las aplicaciones: innumerables y afectan prácticamente a todos los campos de la actividad humana: Medicina, Hogar, Comercio, Educación, Agricultura, Administración, Diseño, Ingeniería, etc...
- Multiproceso.



- Microcomputadora.

Quinta Generación y la Inteligencia Artificial (1981 - Actualidad)

El propósito de la Inteligencia Artificial es equipar a las Computadoras con "Inteligencia Humana" y con la capacidad de razonar para encontrar soluciones. Otro factor fundamental del diseño, la capacidad de la Computadora para reconocer patrones y secuencias de procesamiento que haya encontrado previamente, (programación Heurística) que permita a la Computadora recordar resultados previos e incluirlos en el procesamiento, en esencia, la Computadora aprenderá a partir de sus propias experiencias, usará sus Datos originales para obtener la respuesta por medio del razonamiento y conservará esos resultados para posteriores tareas de procesamiento y toma de decisiones. El conocimiento recién adquirido le servirá como base para la próxima serie de soluciones.

Características Principales:

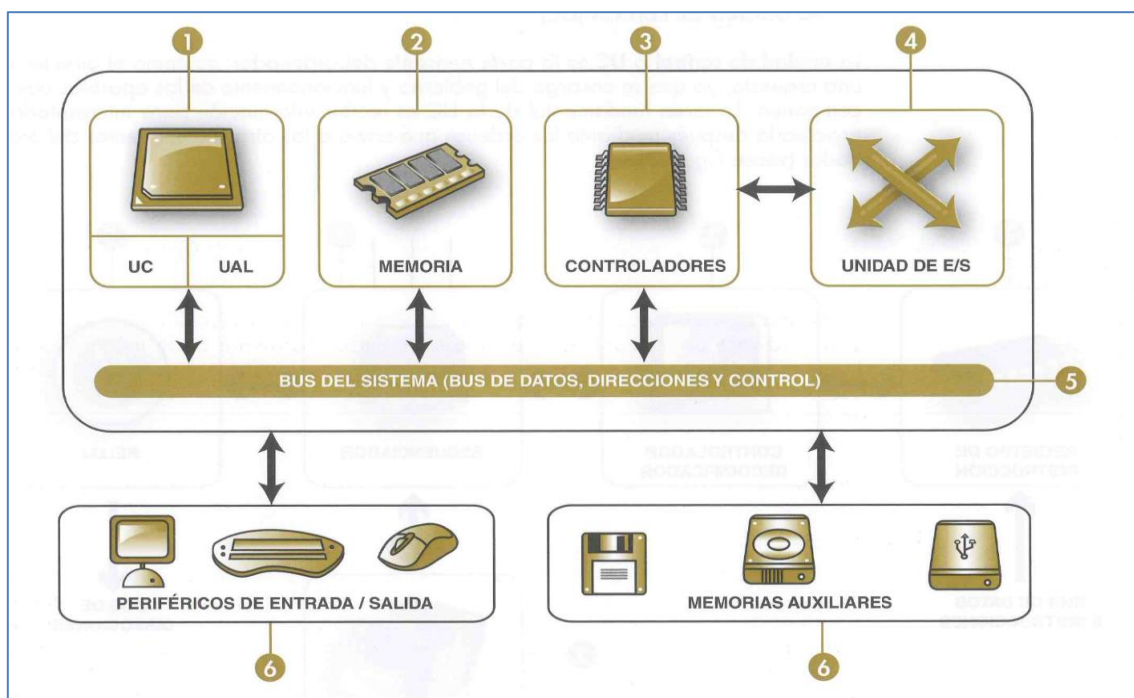
- Mayor velocidad.
- Mayor miniaturización de los elementos.
- Aumenta la capacidad de memoria.
- Multiprocesador (Procesadores interconectados).
- Lenguaje Natural.
- Lenguajes de programación: PROLOG (Programming Logic) y LISP (List Processing).
- Máquinas activadas por la voz que pueden responder a palabras habladas en diversas lenguas y dialectos.

3. COMPONENTES HARDWARE.

Es el conjunto de componentes electrónicos, mecánicos y electromecánicos de un sistema informático. Este concepto hace referencia a todo lo que es material. Todo elemento físico o tangible.

Los componentes físicos se pueden clasificar en:

- ❑ Unidad central de proceso (UCP). Consta de:
 - Unidad aritmético-lógica (UAL).
 - Unidad de control (UC).
- ❑ Memoria central o RAM.
- ❑ Unidad de entrada/salida (E/S).
- ❑ Controladores.
- ❑ Buses.
- ❑ Unidades periféricas o periféricos de entrada/salida.



4. COMPONENTES SOFTWARE.

El software se compone de dos partes fundamentales:

- ✗ El **software básico**: aquella parte del software sin la cual el ordenador no puede funcionar. **SISTEMA OPERATIVO (SO)**.

El SO es el alma del ordenador. Sirve de comunicación entre el usuario y el hardware de la máquina. Controla los recursos hardware de la máquina según las necesidades, los programas de aplicación, el lugar donde se almacenan los datos, el momento en que hay que imprimir, el momento en que se pulsa un botón del ratón.

- ✗ El **software de aplicaciones**: parte software que sirve para procesar la información de forma personalizada. Lo integran los programas y los datos.

Otra clasificación del software de aplicación se hace según este sea estándar o a medida.

- **Estándar**: los programas que se encuentran en el mercado y están a disposición del usuario con unas características predeterminadas. Este software lo utiliza el usuario adaptándolo a su forma de trabajo y a las características del propio software.
- **A Medida**: el que diseñan analistas e implementan (codifican en un lenguaje de programación) programadores atendiendo a las necesidades concretas de los usuarios. En este caso, el software se adapta al usuario.

5. REPRESENTACIÓN DE LA INFORMACIÓN.

La función principal del ordenador es automatizar la información. Tanto para almacenar la información que le suministremos, como para procesarla o transmitirla a otro ordenador, es necesario que la transforme a un sistema que pueda entender.

Internamente el ordenador no trabaja como los seres humanos. No entiende los datos tal y como nosotros se los suministramos. Cualquier dato que le suministremos al ordenador no se almacenará tal y como lo vemos. Cuando pulsamos una tecla, por ejemplo la tecla A o el



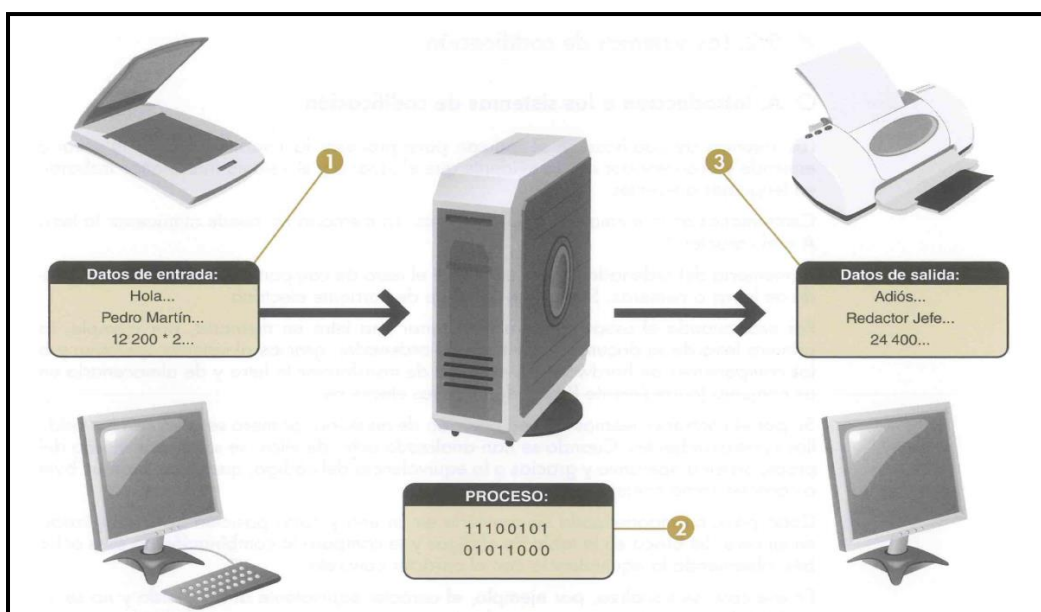
número 9, el ordenador transforma internamente ese dato en un conjunto de 0 y 1. A cada 0 y 1 se le denomina dígito binario o BIT, como veremos más adelante.

Después, si queremos ver los datos que tenemos almacenados en un ordenador, este buscará dónde los tiene almacenados y ese conjunto de 0 y 1 lo convertirá en datos que nosotros sí podamos entender.

5.1. TIPOS DE DATOS.

La primera clasificación que podemos hacer de los tipos de datos es según el lugar que ocupen dentro del proceso:

- ✗ **Datos de entrada.** Son los que se suministran al ordenador desde los periféricos de entrada (teclado, ratón, módem, escáner,...) o desde los diferentes soportes de información (disquetes, discos duros, CD-ROM,...). Forman la primera fase del tratamiento automático de la información: entrada.
- ✗ **Datos intermedios.** Son los que utiliza el ordenador internamente para realizar operaciones antes de mostrar los datos de salida.
- ✗ **Datos de salida.** Son los *resultados* que se obtienen del proceso de automatizar la información. Pueden mostrarse mediante los periféricos de salida o almacenarse en los soportes de información.



Otra clasificación que podemos hacer de los datos, **según varíen o no durante el proceso**, es la siguiente:

- ✗ **Datos fijos.** Permanecen constantes durante el proceso o programa que se les aplique. Reciben el nombre de constantes. Un ejemplo es un programa que emita facturas en euros y pesetas; es evidente que el cambio del euro será siempre el mismo en todo el proceso.
- ✗ **Datos variables.** Se modifican a lo largo del proceso según sucedan determinadas condiciones o acciones realizadas por los programas.



Según la forma de ser utilizados por el ordenador:

- ✗ **Datos numéricos.** Son los dígitos del 0 al 9.
- ✗ **Datos alfabéticos.** Son las letras mayúsculas y minúsculas de la letra A hasta la Z.
- ✗ **Datos alfanuméricos:** Son una combinación de los anteriores, más una serie de caracteres especiales (*, /, -, %, etc.)

Los datos que se procesan en un sistema informático se implementan en códigos numéricos o alfanuméricos para poder utilizarlos.

5.2. SISTEMAS DE NUMERACIÓN.

Se define un **sistema de numeración** como el *conjunto de símbolos y reglas que se utilizan para representar cantidades o datos numéricos*.

Los sistemas de numeración pueden ser:

- ✗ **Posicionales:** el valor de cada símbolo se determina por su valor y la posición que ocupe. El sistema posicional más utilizado a lo largo de la historia es el SISTEMA DECIMAL o en base 10, compuesto por 10 números, del 0 al 9.
- ✗ **No posicionales:** cada símbolo tiene un valor independientemente de la posición donde se encuentre. Los números romanos serían un ejemplo de sistema no posicional. Cada símbolo tiene siempre el mismo valor independientemente del lugar donde se encuentre.

Los sistemas posicionales se caracterizan por la **base** a la que hacen referencia. **Base** es un valor que indica el máximo número de símbolos diferentes que podemos usar en ese sistema para representar cada cantidad numérica.

Todos los sistemas posicionales están basados en el **Teorema Fundamental de la Numeración (TFN)**, que sirve para relacionar una cantidad expresada en cualquier sistema de numeración con la misma cantidad expresada en el sistema decimal.

$$\text{NÚM} = \sum X_i \cdot B^i$$

Donde:

NÚM es el valor de la cantidad que queremos expresar en el sistema de numeración decimal.

X es el valor absoluto del dígito en cuestión.

i es la posición que ocupa el dígito con respecto al punto decimal.

B es la base del sistema de numeración en que está expresado el número.

\sum (sumatorio) indica que para obtener el valor del número (NUM), será necesario sumar todos los productos.

Esta misma fórmula se puede expresar de la siguiente forma:

$$\text{NÚM} = X_n \cdot 10^n + \dots + X_2 \cdot 10^2 + X_1 \cdot 10^1 + X_0 \cdot 10^0 + X_{-1} \cdot 10^{-1} + X_{-2} \cdot 10^{-2} + \dots$$



Por ejemplo el número $3278,52_{(10)}$ puede obtenerse como suma de:

$$\begin{array}{r}
 3000 \\
 200 \\
 70 \\
 8 \\
 0,5 \\
 0,02 \\
 \hline
 3278,52
 \end{array}$$

Por tanto, verifica que:

$$3278,52_{(10)} = 3 \cdot 10^3 + 2 \cdot 10^2 + 7 \cdot 10^1 + 8 \cdot 10^0 + 5 \cdot 10^{-1} + 2 \cdot 10^{-2}$$

Cada posición, por tanto, tiene un peso:

Posición 0 -> Peso B^0

Posición 1 -> Peso B^1

Posición 2 -> Peso B^2

Posición 3 -> Peso B^3

...

Posición -1 -> Peso B^{-1}

Posición -2 -> Peso B^{-2}

ACTIVIDAD 1.1

Utiliza el Teorema Fundamental de la Numeración (TFN), para ver cómo se calcularía el valor de cada dígito de los siguientes números en decimal:

- a) 1785
- b) 345,65
- c) 12,357

5.3. CODIFICACIÓN NUMÉRICA.

Los sistemas de codificación más habituales en un sistema informático son:

- ✖ **Binario.** Utiliza dos símbolos diferentes: el cero y el uno (0, 1). Es el sistema que maneja el ordenador internamente.

Cada uno de los símbolos recibe el nombre de **BIT** (mínima unidad de información posible). VER *Tabla 1.1. Binario-Decimal.*



- ✖ **Octal.** Es un sistema en base 8 que utiliza los símbolos del 0 al 7. El sistema de numeración en base 8 tiene una correspondencia directa con el binario, ya que cada símbolo en base 8 puede representarse mediante una combinación de 3 bits. VER *Tabla 1.2. Binario-Octal.*

Binario	Decimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

Tabla 1.1. Binario-Decimal

Binario	Octal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

Tabla 1.2. Binario-Octal

Un número en octal no podrá tener nunca como dígitos el 8 y el 9. En el sistema octal, al 7 le seguiría el 10, al 17 el 20 y así sucesivamente.

- ✖ **Hexadecimal.** Es un sistema de numeración en base 16. Utiliza símbolos diferentes, del 0 al 9 y las letras A, B, C, D, E y F. Estas letras representan, respectivamente, los dígitos 10, 11, 12, 13, 14 y 15 del sistema decimal. Cada símbolo de base 16 se puede representar mediante una combinación de 4 bits.

Hexadecimal	Decimal	Binario
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100



D	13	1101
E	14	1110
F	15	1111

Tabla 1.3. Hexadecimal-Decimal-Binario.

5.4. CAMBIOS DE BASE DE NUMERACIÓN.

El sistema informático trabaja en el sistema de numeración binario. Nosotros trabajamos en el sistema de numeración decimal. El ordenador no entiende el sistema de numeración decimal para realizar sus cálculos, pero nosotros no entendemos el binario para realizar nuestros cálculos.

Es necesario saber interpretar el código binario para poder entender las operaciones que en muchas ocasiones se realizan dentro del ordenador. Para ello, debemos aprender a pasar números binarios a decimales y a la inversa. Por extensión, el ordenador utiliza los sistemas de numeración de base 8 y base 16 (por ser múltiplos del sistema binario) para mostrarnos información relativa a algunos procesos que realiza.

5.4.1. CAMBIOS DE BINARIO A DECIMAL

Utilizando el Teorema fundamental de la numeración, se multiplica el valor de cada dígito binario por la potencia de 2 de la posición que ocupe el dígito dentro del número. Por ejemplo:

$$110100_2 = 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 2^5 + 2^4 + 2^2 = 52_{10}$$

Primero se toman los dígitos binarios, 6 en total, y se van multiplicando por potencias de 2 de izquierda a derecha.

ACTIVIDAD 1.2

Transforma los siguientes números en binario a decimal:

- a) 11011
- b) 0,10100
- c) 10100,001

5.4.2. CAMBIOS DE DECIMAL A BINARIO.

Para pasar la parte entera, de decimal a binario, deberemos dividir entre 2 hasta que resulte el cociente 0. El número binario se obtiene uniendo todos los restos en orden inverso.

DIVIDENDO	DIVISOR
RESTO	COCIENTE



Pasar $90_{(10)}$ a base 2:

1. Dividimos el número por 2.

$$\begin{array}{r} 90 \overline{) 2} \\ 0 \quad 45 \end{array}$$

2. El cociente que obtenemos lo dividimos de nuevo por 2.

$$\begin{array}{r} 90 \overline{) 2} \\ 0 \quad 45 \overline{) 2} \\ 1 \quad 22 \end{array}$$

3. El nuevo cociente lo volvemos a dividir por 2.

$$\begin{array}{r} 90 \overline{) 2} \\ 0 \quad 45 \overline{) 2} \\ 1 \quad 22 \overline{) 2} \\ 0 \quad 11 \end{array}$$

4. Y así sucesivamente hasta que aparezca un cociente igual a 0.

$$\begin{array}{r} 90 \overline{) 2} \\ 0 \quad 45 \overline{) 2} \\ 1 \quad 22 \overline{) 2} \\ 0 \quad 11 \overline{) 2} \\ 1 \quad 5 \overline{) 2} \\ 1 \quad 2 \overline{) 2} \\ 0 \quad 1 \overline{) 2} \\ 1 \quad 0 \end{array}$$

5. Ordenamos los restos sucesivos que aparecen en las divisiones, pero en orden inverso, y obtenemos la nueva codificación en base 2:

$$\begin{array}{r} 90 \overline{) 2} \\ 0 \quad 45 \overline{) 2} \\ 1 \quad 22 \overline{) 2} \\ 0 \quad 11 \overline{) 2} \\ 1 \quad 5 \overline{) 2} \\ 1 \quad 2 \overline{) 2} \\ 0 \quad 1 \overline{) 2} \\ 1 \quad 0 \end{array}$$

A red arrow points from the bottom right (1 0) up to the top left (0), indicating the reverse order of the remainders.

6. Resultado: $90_{(10)} = 1011010_{(2)}$

Para convertir la parte fraccionaria del número decimal a binario se haría multiplicando sucesivamente por 2 la parte fraccionaria del número decimal de partida y las partes fraccionarias que se van obteniendo en los productos sucesivos. El número binario se forma con las partes enteras (que serán ceros y unos) de los productos obtenidos.



Transformar a binario natural el número decimal 0,1875

$$0,1875 * 2 = 0,3750$$

$$0,3750 * 2 = 0,7500$$

$$0,7500 * 2 = 1,5000$$

$$0,5000 * 2 = 1,0000$$

El resultado final sería el siguiente:

$$0,1875_{(10)} = 0,0011_{(2)}$$

ACTIVIDAD 1.3

Pasar a binario los siguientes números decimales:

- a) 26
- b) 32,375
- c) 74,250
- d) 50,875

5.4.3. CAMBIOS DE BINARIO A OCTAL

Cada número en octal tiene una correspondencia directa con el número en binario mediante un conjunto de 3 bits. Sólo es necesario observar la Tabla 1.2.

Transformar a octal el número binario 11011101

Para pasar de binario a octal debemos coger los números binarios en grupos de 3, desde la coma hacia la izquierda, para la parte entera, y desde la coma hacia la derecha, para la parte fraccionaria. Si hubiera que añadir ceros para formar el conjunto de 3, se añadiría en el lugar donde no alterase el resultado del número, es decir, en la parte entera a la izquierda y en la parte fraccionaria a la derecha.

1. Agrupo en grupos de 3 bits de derecha a izquierda -> **011 | 011 | 101**
2. Miro la tabla 1.2 para ver su equivalente

$$011 = 3$$

$$011 = 3$$

$$101 = 5$$

3. Resultado: $11011101_{(2)} = 335_{(8)}$



5.4.4. CAMBIOS DE OCTAL A BINARIO.

Transformar a binario el número 34,03 en octal

Para pasar de octal a binario, nos fijamos en la tabla 1.2 y la conversión es directa, convirtiendo cada número en octal a 3 números en binario, incluidos los ceros.

1. 3 = 011
2. 4 = 100
3. 0 = 000
4. 3 = 011
5. Resultado: $34,03_8 = 011100,000011_2$

ACTIVIDAD 1.4

Pasar el siguiente número octal a binario:

- a) 1342

Calcula para el siguiente número en binario su valor en el sistema octal:

- a) 10001,1001

5.4.5. CAMBIOS DE OCTAL A DECIMAL

Para pasar un número de octal a decimal aplicamos la fórmula del **TFN** siendo en este caso $B=8$.

Pasar el número octal 1367,25₈ a decimal:

$$1367,25 = 1 \cdot 8^3 + 3 \cdot 8^2 + 6 \cdot 8^1 + 7 \cdot 8^0 + 2 \cdot 8^{-1} + 5 \cdot 8^{-2} = 759,328125_{10}$$

5.4.6. CAMBIOS DE BINARIO A HEXADECIMAL

Cada número en hexadecimal tiene una correspondencia directa con el número binario mediante un conjunto de 4 bits. Sólo es necesario observar la tabla 1.3.

Pasar el número binario 11010 a hexadecimal:

Para pasar de binario a hexadecimal debemos coger los números binarios en grupos de 4, desde la coma hacia la izquierda, para la parte entera, y desde la coma hacia la derecha, para la parte fraccionaria. Si hubiera que añadir ceros para formar el conjunto de 4, se añadirían en el lugar donde no alterase el resultado del número, es decir, en la parte entera a la izquierda y en la parte fraccionaria a la derecha.

1. Agrupo en grupos de 4 bits de derecha a izquierda -> 0001 | 1010
2. Miro la tabla 1.3 para ver su equivalente: 0001 = 1 y 1010 = A
3. Resultado $11010_2 = 1A_{16}$



5.4.7. CAMBIOS DE HEXADECIMAL A BINARIO

Transformar a binario el número 3AF₍₁₆₎

Para pasar de hexadecimal a binario, nos fijamos en la tabla 1.3 y la conversión es directa, convirtiendo cada número en hexadecimal a 4 números en binario, incluidos los ceros.

1. 3 = 0011
2. A = 1010
3. F = 1111
4. **Resultado: 3AF₍₁₆₎ = 001110101111₍₂₎**

ACTIVIDAD 1.5

Pasar el siguiente número hexadecimal a binario:

- a) F0,0C

Calcula para los siguientes números en binario su valor en el sistema octal:

- b) 101011,010001
- c) 110110010

5.4.8. CAMBIOS DE HEXADECIMAL A DECIMAL

Para transformar un número hexadecimal a decimal aplicamos la expresión del **TFN** con B=16.

Pasar el número hexadecimal A798C,1E₍₁₆₎ a decimal.

$$10 \cdot 16^4 + 7 \cdot 16^3 + 9 \cdot 16^2 + 8 \cdot 16^1 + 12 \cdot 16^0 + 1 \cdot 16^{-1} + 14 \cdot 16^{-2} = 686476,1171_{(10)}$$

5.5. CODIFICACIÓN ALFANUMÉRICA

Ya sabemos que los datos, además de ser numéricos, pueden ser alfabéticos o alfanuméricos. Normalmente, con los datos alfanuméricos podemos construir instrucciones y programas. Por otro lado, es lógico pensar que el ordenador no solamente procesará datos numéricos, sino también datos alfabéticos y combinaciones de los anteriores, como datos alfanuméricos.

Los sistemas de codificación alfanumérica sirven para representar una cantidad determinada de símbolos en binario. A cada símbolo le corresponderá una combinación de un número de bits.

Los sistemas de codificación más importantes son:



- ✗ **ASCII** (American Standard Code for Information Interchange). Este sistema utiliza una combinación de 7 u 8 bits, dependiendo del fabricante, para representar cada símbolo. Es el más utilizado y el que emplea símbolos diferentes. Con este código se pueden representar dígitos del 0 al 9, letras mayúsculas de la A a la Z, letras minúsculas, caracteres especiales y algunos otros denominados de control.
- ✗ **BCD (Binary Code Decimal)** o Decimal codificado binario fue uno de los primeros utilizados. Representaba caracteres alfanuméricos en conjuntos de 6 bits. También se usó para codificar los dígitos numéricos en binario empleando 4 bits. De forma que la codificación del dígito era directa a su valor en binario.
- ✗ **EBCDIC (Extended BCD Interchange Code)**. Cada símbolo se representa por una combinación de 8 bits agrupados en dos bloques de 4. Es el formato extendido de BCD.
- ✗ **FIELDATA**. Este código se utilizó en sistemas militares, sobre todo en EE.UU. para la transmisión de datos. Utilizaba para ello 6 bits.
- ✗ **UNICODE**. Es el estándar más utilizado en la actualidad. Permite que un mismo texto o página Web se pueda visualizar sin problemas en la mayoría de los idiomas y en diferentes plataformas.
En Unicode podemos utilizar tres tipos de codificación: UTF-8, UTF-16 y UTF-32.

Tabla 1.4. Esquema de la tabla de código ASCII de 8 bits hasta el carácter 127.

Caracteres no imprimibles				Caracteres imprimibles											
Nombre	Dec	Hex	Car.	Dec	Hex	Car.	Dec	Hex	Car.	Dec	Hex	Car.	Dec	Hex	Car.
Nulo	0	00	NUL	32	20	Espacio	64	40	@	96	60	`			
Inicio de cabecera	1	01	SOH	33	21	!	65	41	A	97	61	a			
Inicio de texto	2	02	STX	34	22	"	66	42	B	98	62	b			
Fin de texto	3	03	ETX	35	23	#	67	43	C	99	63	c			
Fin de transmisión	4	04	EOT	36	24	\$	68	44	D	100	64	d			
enquiry	5	05	ENQ	37	25	%	69	45	E	101	65	e			
acknowledge	6	06	ACK	38	26	&	70	46	F	102	66	f			
Campanilla (beep)	7	07	BEL	39	27	'	71	47	G	103	67	g			
backspace	8	08	BS	40	28	(72	48	H	104	68	h			
Tabulador horizontal	9	09	HT	41	29)	73	49	I	105	69	i			
Salto de línea	10	0A	LF	42	2A	*	74	4A	J	106	6A	j			
Tabulador vertical	11	0B	VT	43	2B	+	75	4B	K	107	6B	k			
Salto de página	12	0C	FF	44	2C	,	76	4C	L	108	6C	l			
Retorno de carro	13	0D	CR	45	2D	-	77	4D	M	109	6D	m			
Shift fuera	14	0E	SO	46	2E	.	78	4E	N	110	6E	n			
Shift dentro	15	0F	SI	47	2F	/	79	4F	O	111	6F	o			
Escape línea de datos	16	10	DLE	48	30	0	80	50	P	112	70	p			
Control dispositivo 1	17	11	DC1	49	31	1	81	51	Q	113	71	q			
Control dispositivo 2	18	12	DC2	50	32	2	82	52	R	114	72	r			
Control dispositivo 3	19	13	DC3	51	33	3	83	53	S	115	73	s			
Control dispositivo 4	20	14	DC4	52	34	4	84	54	T	116	74	t			
neg acknowledge	21	15	NAK	53	35	5	85	55	U	117	75	u			
Sincronismo	22	16	SYN	54	36	6	86	56	V	118	76	v			
Fin bloque transmitido	23	17	ETB	55	37	7	87	57	W	119	77	w			
Cancelar	24	18	CAN	56	38	8	88	58	X	120	78	x			
Fin medio	25	19	EM	57	39	9	89	59	Y	121	79	y			
Sustituto	26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z			
Escape	27	1B	ESC	59	3B	;	91	5B	[123	7B	{			
Separador archivos	28	1C	FS	60	3C	<	92	5C	\	124	7C				
Separador grupos	29	1D	GS	61	3D	=	93	5D]	125	7D	}			
Separador registros	30	1E	RS	62	3E	>	94	5E	^	126	7E	~			
Separador unidades	31	1F	US	63	3F	?	95	5F	_	127	7F	DEL			



6. OPERACIONES EN BINARIO

Al igual que ocurre con el sistema decimal, en el sistema binario podemos realizar las operaciones aritméticas: suma, resta, multiplicación y división.

6.1. SUMA Y RESTA EN BINARIO

La suma binaria es parecida a la suma en decimal, con la diferencia de que se manejan sólo dos dígitos. Si el resultado de la suma excede de 1, se agrega un acarreo a la suma parcial siguiente. Para realizar sumas nos fijaremos en las tablas siguientes.

SUMA BINARIA
$0 + 0 = 0$
$0 + 1 = 1$
$1 + 0 = 1$
$1 + 1 = 0$, acarreo 1

RESTA BINARIA
$0 - 0 = 0$
$0 - 1 = 1$, acarreo 1 que se suma al siguiente sustraendo.
$1 - 0 = 1$
$1 - 1 = 0$



a) Suma sin acarreos:

			1	0	0	0	0	→ 16
+		1	0	1	0	0	1	→ 41
		1	1	1	0	0	1	→ 57

b) Suma con acarreos:

				1	1	1		Acarreos
				↓	↓	↓		
	1	0	1	0	1	1	1	→ 87
+		1	0	0	0	0	1	→ 33
	1	1	1	1	0	0	0	→ 120

Cuando nos encontramos con tres unos, la suma da 1 y de acarreo 1.

				1	1	1		Acarreos
				↓	↓	↓		
		1	1	0	1	1	1	→ 55
+		1	0	0	0	1	1	→ 35
	1	0	1	1	0	1	0	→ 90

c) Sumas con decimales:

				1	1	1	1		1	Acarreos
				↓	↓	↓	↓		↓	
				1	1	0	,	1	1	→ 6,75
+				1	0	1	,	0	1	→ 5,25
	1	1	0	0	,	0	0			→ 12,00

d) Resta sin acarreos:

		1	1	1	0	1	0	1	→ 117
+		1	0	0	0	0	1		→ 33
		1	0	1	0	1	0	0	→ 84



e) Resta con acarreos:

- Cuando nos encontramos con el primer 0 - 1, el resultado es 1 y nos llevamos 1, que sumaremos al siguiente sustraendo.
- Si al sumar nos volvemos a llevar 1 (caso de sumar 1 de acarreo + 1 en sustraendo), ese 1 pasa al siguiente sustraendo, y así sucesivamente hasta que dé 0.

$$\begin{array}{r}
 \begin{array}{ccccccc}
 1 & 1 & 1 & 0 & 1 & 0 & 1 \\
 - & \downarrow 1 & \downarrow 1 & \downarrow 1 & & \downarrow 1 & \\
 \hline
 0 & 1 & 1 & 1 & 0 & 1 & 1
 \end{array} \begin{array}{l} \rightarrow 117 \\ \text{Acarreos} \\ \rightarrow 58 \\ \rightarrow 59 \end{array} \\
 \\
 \begin{array}{ccccccc}
 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
 - & & \downarrow 1 & \downarrow 1 & & \downarrow 1 & \\
 \hline
 1 & 0 & 0 & 1 & 0 & 1 & 0
 \end{array} \begin{array}{l} \rightarrow 101 \\ \text{Acarreos} \\ \rightarrow 27 \\ \rightarrow 74 \end{array}
 \end{array}$$

f) Resta con decimales:

$$\begin{array}{r}
 \begin{array}{ccccccc}
 1 & 0 & 0 & 0 & 1 & , & 0 & 1 \\
 - & \downarrow 1 & \downarrow 1 & \downarrow 1 & \downarrow 1 & & \downarrow 1 & \\
 \hline
 0 & 0 & 1 & 0 & 1 & , & 1 & 1
 \end{array} \begin{array}{l} \rightarrow 17,25 \\ \text{Acarreos} \\ \rightarrow 11,75 \\ \rightarrow 5,5 \end{array}
 \end{array}$$

6.2. MULTIPLICACIÓN EN BINARIO.

Se realiza como en la multiplicación decimal, con la diferencia de que luego se hacen las sumas en binario. Para los productos, utilizaremos la siguiente tabla.

Multiplicación binaria

$$\begin{array}{l}
 0 \cdot 0 = 0 \\
 0 \cdot 1 = 0 \\
 1 \cdot 0 = 0 \\
 1 \cdot 1 = 1
 \end{array}$$

a) Multiplicar 25 (11001) por 5 (101).

$$\begin{array}{r}
 \begin{array}{ccccccc}
 & & 1 & 1 & 0 & 0 & 1 \\
 \times & & & & 1 & 0 & 1 \\
 \hline
 & & 1 & 1 & 0 & 0 & 1 \\
 + & & 0 & 0 & 1 & 0 & 0 \\
 + & 1 & 1 & 0 & 0 & 1 & 0 \\
 \hline
 1 & 1 & 1 & 1 & 1 & 0 & 1
 \end{array} \begin{array}{l} \rightarrow 25 \\ \rightarrow 5 \\ \rightarrow 125 \end{array}
 \end{array}$$

b) Multiplicar 23 (10111) por 14 (1110). Nos encontramos con columnas en las que hay que sumar cuatro 1:

$$\begin{array}{r}
 \begin{array}{ccccccc}
 & & 1 & 0 & 1 & 1 & 1 \\
 \times & & & 1 & 1 & 1 & 0 \\
 \hline
 & & 1 & 0 & 1 & 1 & 1 \\
 + & & 0 & 0 & 1 & 1 & 1 \\
 + & 1 & 0 & 1 & 1 & 1 & 1 \\
 \hline
 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0
 \end{array} \begin{array}{l} \rightarrow 23 \\ \rightarrow 14 \\ \rightarrow 322 \end{array}
 \end{array}$$



Tabla 5.4. Equivalencias de medidas de información.

ACTIVIDAD 1.6

Busca en Internet otras medidas de información.

El número 1024 es una potencia de 2 (2^{10}). Su uso está justificado, ya que el ordenador utiliza internamente el sistema de codificación binario para todas sus operaciones.

Actualmente, la capacidad de la memoria RAM se mide en Mb o Gb, y la capacidad de los discos duros en Gb o Tb.

FIN UNIDAD 1: INTRODUCCIÓN A LOS SISTEMAS INFORMÁTICOS