

Sistemas Informáticos, 1º DAM

Representación de la Información

Datos e información

Según el diccionario de la RAE un **ordenador** o computadora electrónica es una máquina que, mediante determinados programas, almacena, trata la información y resuelve problemas de diversa índole. Así pues, un ordenador recibe y procesa conjuntos de datos para procesarlos y producir información útil para el usuario.

Los datos suelen ser representaciones simbólicas (nombres, imágenes, olores, sonidos, colores, etc.) de un atributo o característica. Por sí solos, los datos no tienen sentido alguno, para ello es necesario procesarlos y obtener así lo que se denomina información.

Así, podemos definir la información, como un conjunto organizado de datos que proporcionan un significado o sentido a las cosas.

Se define **código** como un conjunto de símbolos que pueden combinarse entre sí para representar cierta información. El proceso de traducción de un código origen a otro destino se denomina codificación, y la tarea inversa decodificación. Como ejemplo podemos proponer el código Morse, creado por Alfred Vail en 1835 mientras colaboraba con Samuel Morse en la invención del telégrafo. Vail creó un método según el cual cada letra o símbolo era transmitido de forma individual con un código consistente en rayas y puntos.

C ó d i g o		M o r s e		I n t e r n a c i o n a l			
A	• —	N	• —	1	• — — — —	periodo	• • • • •
B	• • • •	O	— — — —	2	• • — — —	coma	— — • • — —
C	• — • •	P	• — — •	3	• • • — —	dos puntos	— — — — • •
D	• • •	Q	• — — • —	4	• • • • —	pregunta	• • — — • •
E	•	R	• — •	5	• • • • •	apóstrofe	• — — — — •
F	• • • •	S	• • •	6	• • • • •	guión	— • • • • •
G	• — •	T	—	7	• — • • •	fracción	— • • • •
H	• • • •	U	• • —	8	• — — • •	paréntesis	— • — — — •
I	• •	V	• • • —	9	• — — — •	comillas	• • • • •
J	• — — — —	W	• — • —	0	— — — — —		
K	• • —	X	• — • •				
L	• — • •	Y	• — • — —				
M	— —	Z	— • — •				

Usando esta codificación podemos representar el término INFORMATICA con los siguientes puntos y rayas:

• • — • • • — • — — — • — • — — • — — • • — • — • • —
I N F O R M A T I C A

Codificación de la información

La mayoría de las cosas que podemos medir cuantitativamente aparecen en la naturaleza en forma analógica, como por ejemplo la temperatura, que varía dentro de un rango continuo de valores. Sin embargo, un ordenador es un dispositivo digital, esto es, procesa la información como un conjunto de valores discretos.

La diferencia entre "valores continuos" y "valores discretos" es que entre un valor discreto y el siguiente valor discreto hay un **salto**, cosa que no ocurre entre valores continuos sucesivos.

Como ejemplo un velocímetro analógico varía entre un rango de valores continuos, mientras que el velocímetro digital lo hace entre un rango de valores discretos. En el caso del velocímetro digital de la imagen el **salto** entre valores consecutivos es de 0,1 Km/h (valores discretos). Lo que no ocurre con el analógico (valores continuos).



En los sistemas digitales, aunque a menudo veamos números en decimal (diez dígitos), internamente se utiliza numeración binaria, únicamente dos dígitos, las combinaciones de estos dos posibles estados se utilizan para representar números, símbolos, caracteres alfabéticos y otro tipo de datos, dando lugar a lo que se conoce como código binario.

Este código binario define la unidad mínima de información que podemos codificar en un sistema digital: el **bit** (Binary Digit), que puede tomar los valores 0 y 1, permitiéndonos representar dos valores cualesquiera. Imaginemos, por ejemplo, que queremos representar los dos posibles estados de una bombilla: encendido o apagado. Nos bastaría con un único bit.



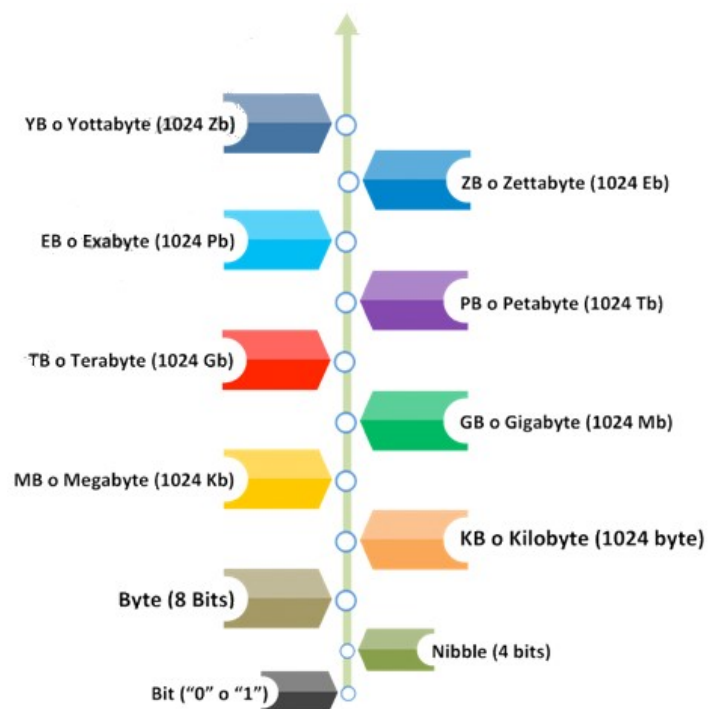
Parece evidente que un bit no es suficiente para representar cualquier tipo de información, ya que con él podemos representar dos posibles estados. Para solucionar este problema tendremos que utilizar agrupaciones de bits. Supongamos ahora que tenemos dos bits. ¿Cuántos posibles estados podríamos representar con ellos? Imaginemos que tenemos dos bombillas, una junto a la otra: ¿cuántas combinaciones encendido/apagado podríamos tener? Efectivamente: cuatro.



Según lo anterior, cuanto mayor sea la agrupación de bits, mayor será la cantidad de información que podremos codificar. Se conoce como **Byte** a una agrupación de 8 bits. Dada la tendencia al almacenamiento masivo de información, se hace necesario utilizar unidades equivalentes, esto es, múltiplos del byte.

El **Nibble** es una agrupación de 4 bits muy utilizada en los años 70 cuando surgieron los primeros ordenadores, ya que éstos trabajaban con instrucciones constituidas por grupos de 4 bits. Utilizaremos también este término cuando hablemos del sistema de codificación BCD y, más ampliamente para referirse a las agrupaciones de 4 bits que componen los dígitos hexadecimales

Como veremos más adelante, un byte es suficiente para representar cualquier letra de nuestro alfabeto. Teniendo esto en cuenta, si



decimos que un *pendrive* tiene una capacidad de 32 Gigabytes, en realidad estamos diciendo que puede almacenar... ¡32.000.000.000 de caracteres!

Sistemas de numeración

Se define un sistema de numeración como un conjunto de reglas que permiten, con una cantidad finita de símbolos, representar cualquier número. Estamos habituados a trabajar de forma natural y a diario con el sistema de numeración decimal o arábigo para representar cualquier número. Éste cuenta con un total de diez símbolos (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) que, combinándolos según determinadas reglas, nos permiten representar cualquier cifra numérica.

De esta manera, combinando el 2 y el 5 podemos construir las siguientes cifras numéricas: 25, 52. Pero, detengámonos un momento y observemos dichas cantidades. ¿Tiene el 2 el mismo significado en ambas? La respuesta debería ser inmediata: no. En ambas cifras, el 2 tiene un peso múltiplo de 10 y de 1, respectivamente.

En el sistema decimal, la posición relativa de un dígito en una cifra determina el valor de la misma, esto es, cada dígito tiene un peso que depende de la posición que ocupa y aumenta de derecha a izquierda. Los sistemas de numeración que verifican dicha propiedad se conocen como posicionales. Existen también sistemas no posicionales, como el sistema de numeración romano.

Todo sistema de numeración se caracteriza además por su base, que determina el número de símbolos con el que contamos para construir cualquier cifra. Con todo esto, el teorema fundamental de la numeración determinará la manera en que podemos construir un número.

$$6259 = 6 \cdot 10^3 + 2 \cdot 10^2 + 5 \cdot 10^1 + 9 \cdot 10^0$$

Sistema de numeración binario

El sistema de numeración binario es un sistema en base 2 que utiliza únicamente los dígitos 0 y 1 para representar cualquier cantidad numérica. Pero, ¿cómo lo hace? Pues de igual forma como lo hacemos en el sistema decimal. Comenzamos en cero y escribimos hasta el nueve antes de quedarnos sin dígitos. 0 1 2 3 4 5 6 7 8 9

Continuamos utilizando dos dígitos, contamos hasta 99 y, llegados a este punto se hará necesario un tercer dígito para contar hasta 999, y así sucesivamente, hasta el infinito y más allá.

En binario sucede igual. Empezamos a contar utilizando los dígitos 0 y 1, pero en seguida nos quedamos sin posibilidades, a no ser que utilicemos dos dígitos. Tal y como vimos anteriormente, ahora tenemos cuatro posibilidades: 00, 01, 10, 11. Con tres dígitos podemos seguir contando, luego cuatro y así sucesivamente. Antes de continuar, ¿serías capaz de contar en binario desde cero hasta quince? En primer lugar debemos determinar cuántos dígitos binarios o bits necesitamos. Rápidamente observamos que necesitamos combinar 4 bits para contar de 0 a 15. **En general, con n bits podemos contar hasta 2^n .**

Número decimal	Número binario				Número decimal	Número binario			
0	0	0	0	0	8	1	0	0	0
1	0	0	0	1	9	1	0	0	1
2	0	0	1	0	10	1	0	1	0
3	0	0	1	1	11	1	0	1	1
4	0	1	0	0	12	1	1	0	0
5	0	1	0	1	13	1	1	0	1
6	0	1	1	0	14	1	1	1	0
7	0	1	1	1	15	1	1	1	1

Obviamente, cuanto mayor sea la cantidad de números que queramos representar, mayor será la cantidad de bits necesaria para ello. De esta manera, necesitaremos 5 bits para escribir los números del 0 al 31, 6 bits si queremos llegar al 64, y así sucesivamente. Pero, ¿cuántos bits serían necesarios para representar el número 12.949? Obtenemos un resultado aproximado a nuestra pregunta calculando el $\log_2 12.949$ y redondeando el resultado al siguiente entero.

$$\log_2 12.949 = \log 12.949 / \log 2 \approx 14$$

Esto es, necesitaremos exactamente 14 bits, ya que $2^{14} = 16.384$. No podríamos representar el valor que queríamos si optásemos por 13 bits, ya que con éstos sólo podemos escribir los valores comprendidos el 0 y el 8.192.