

# **Preparador Informática**

[www.preparadorinformatica.com](http://www.preparadorinformatica.com)

**TEMA 1. INFORMÁTICA / S.A.I.**

**REPRESENTACIÓN Y COMUNICACIÓN  
DE LA INFORMACIÓN**

## **TEMA 1 INF / SAI: REPRESENTACIÓN Y COMUNICACIÓN DE LA INFORMACIÓN**

### **1. INTRODUCCIÓN**

### **2. REPRESENTACIÓN DE LA INFORMACIÓN**

#### **2.1. REPRESENTACIÓN DE TEXTOS**

#### **2.2. REPRESENTACIÓN DE SONIDOS**

#### **2.3. REPRESENTACIÓN DE IMÁGENES**

#### **2.4. REPRESENTACIÓN DE NÚMEROS**

##### **2.4.1. NÚMEROS ENTEROS**

##### **2.4.2. NÚMEROS REALES**

### **3. COMUNICACIÓN DE LA INFORMACIÓN**

#### **3.1. ENCRIPCIÓN DE DATOS**

#### **3.2. COMPRESIÓN DE DATOS**

#### **3.3. REDUNDANCIA DE DATOS**

### **4. RECURSOS Y HERRAMIENTAS EDUCATIVAS DE INTERÉS**

### **5. CONCLUSIÓN**

### **6. BIBLIOGRAFÍA**

## 1. INTRODUCCIÓN

Un ordenador es una máquina que procesa, memoriza y transmite información. Esta información se da en la forma de datos, que son conjuntos de símbolos utilizados para expresar o representar un valor numérico, un hecho, un objeto o una idea; en la forma adecuada para ser objeto de tratamiento.

La ejecución de un programa en un ordenador implica la realización de unos tratamientos con unos datos. Para que el ordenador pueda ejecutar un programa es necesario darle dos tipos de informaciones: las **instrucciones** que forman dicho programa y los **datos** con los que debe operar. La elección de qué tipos de datos e instrucciones se pueden utilizar, así como qué representación van a tener, son las decisiones más importantes y decisivas que deben tomar los diseñadores de un nuevo computador, ya que son clave para poder diseñar sus componentes estructurales.

El presente tema está dedicado a estudiar los aspectos relacionados con la representación de la información en el interior de los computadores y también a describir el proceso de comunicación de la información, describiendo aspectos tales como el cifrado de datos, técnicas para la detección de errores y técnicas para la compresión de datos.

## El ordenador (informática)

## 2. REPRESENTACIÓN DE LA INFORMACIÓN

Relacionado con cómo se representa la información con la que trabaja un computador podemos considerar que existen dos niveles en la representación de la información:

- Nivel de **representación externa**: usada por las personas e inadecuada para el ordenador.
- Nivel de **representación interna**: adecuada al ordenador y no inteligible directamente para las personas.

A continuación, se describe cómo la información expresada en el nivel de representación externa es representable en los computadores (representación

interna). Este paso de una representación a otra se denomina codificación y el proceso inverso decodificación.

Los computadores internamente representan la información usando una lógica binaria, es decir, su alfabeto tiene únicamente dos símbolos: 0 y 1. Esto se debe a que son más fiables y más sencillos de construir al contemplar únicamente dos valores.

Dependiendo del tipo de información que se represente se utilizarán diferentes representaciones. Se pueden diferenciar cuatro tipos de representaciones:

- Representación de textos
- Representación de sonidos
- Representación de imágenes
- Representación de datos numéricos

## 2.1. REPRESENTACIÓN DE TEXTOS

La información se suele introducir en el computador utilizando el lenguaje escrito cuyas unidades básicas son caracteres. Los caracteres se suelen agrupar en 5 grupos:

- **Caracteres alfabéticos:** corresponde a las letras mayúsculas y minúsculas del abecedario inglés.

$a, b, \dots, z, A, B, \dots, Z$

- **Caracteres numéricos:** corresponde a las diez cifras decimales.

$0, 1, \dots, 9$

- **Caracteres especiales:** son símbolos ortográficos y matemáticos no incluidos en los grupos anteriores.

$, (, ), \{, \}, +, -, \dots, \text{etc.}$

- **Caracteres gráficos:** son símbolos con los que se pueden representar figuras, dibujos, etc.

♦ ♥ ♠ ♣



- **Caracteres de control:** se corresponden con las órdenes de control (carácter de salto de página, comienzo de línea, etc.)

Debido a que en la representación interna de la información en los computadores solo disponemos de ceros y unos, la comunicación hombre-máquina se debe establecer a través de una correspondencia entre el conjunto de todos los caracteres,  $\alpha = \{a, \dots, z, A, \dots, Z, 0, \dots, 9\}$ , y un conjunto  $\beta = \{0, 1\}^n$ .

Esta correspondencia establece lo que se conoce como código de entrada/salida. Al proceso de asignar a cada uno de los caracteres una secuencia de ceros y unos se le denomina codificación. En principio, esta correspondencia es arbitraria, pero existen códigos de E/S normalizados que son utilizados por diferentes constructores de computadores.

Algunos de los códigos entrada/salida más utilizados son:

- **EBCDIC** (*Extended Binary Coded Decimal Interchange Code*): utiliza 8 bits para representar cada carácter, por lo que puede representar hasta 256 caracteres.
- **ASCII** (*American Standard Code for Information Interchange*): su versión básica utiliza 7 bits y es capaz de representar hasta 128 caracteres. Existen versiones ampliadas que utilizan 8 bits. Entre ellas se encuentran los códigos ISO 8859-n, donde n es un número que identifica el juego de los nuevos caracteres introducidos. Por ejemplo, la **ISO 8859-1** (Latín-1) se proyectó para América y Europa occidental para incluir las vocales con tilde y otras letras latinas no usadas en los países anglosajones.
- **UNICODE:** creado en 1991 con el objetivo de solucionar los problemas relacionados con las limitaciones de los códigos existentes hasta ese momento. Existen diferentes formas de codificar Unicode: UTF-8, UTF-16 y UTF-32. Esta última utiliza todos los puntos posibles de Unicode usando 32 bits (4 bytes por carácter). UTF-16 y UTF-8 son codificaciones de longitud variable. Esto significa que, si un carácter se puede representar con un sólo byte, UTF-8 empleará sólo un byte. Si requiere dos bytes, usará dos, y así sucesivamente.

## 2.2. REPRESENTACIÓN DE SONIDOS

Una señal de sonido se capta por medio de un micrófono que produce una señal analógica. Para poder almacenarla y procesarla utilizando técnicas digitales se realiza un proceso de muestreo en el intervienen dos parámetros fundamentales:

- **Frecuencia de muestreo:** cuanto mayor sea la frecuencia a la que se muestrea mayor será la calidad del sonido grabado.
- **Precisión:** es el número de bits con el que se representa cada muestra.

Una vez obtenidos los valores binarios de las muestras hay que codificarlos de acuerdo con un determinado formato. Hay diversos formatos de codificación, entre los que se encuentran:

- PCM (Pulse Code Modification)
- DPCM (Differential Pulse Code Modulation)
- ADPCM (Adaptive Differential Pulse Modulation)
- $\mu$  – law
- MPEG
- Etc.

Cuanto mayor sea la frecuencia y la precisión mayor será la calidad del sonido, pero también será mayor el espacio de almacenamiento ocupado.

## 2.3. REPRESENTACIÓN DE IMÁGENES

Existen dos formas básicas de representar las imágenes:

### A. Mapas de bits

La imagen está compuesta por puntos (píxeles), y a cada uno de ellos se le puede asociar su nivel de color. Para codificar y almacenar la imagen hay que tener en cuenta dos factores: número de píxeles y nivel de color asociado a cada pixel. La calidad de una imagen viene determinada por la resolución de la imagen (número de píxeles por línea x número de píxeles por columna).

Algunos de los principales formatos son: BMP, TIFF, JPEG, GIF, PNG, etc.

## B. Mapas de vectores:

La imagen se descompone en una colección de objetos tales como líneas, polígonos y textos con sus respectivos atributos (grosor, color, etc.). Se utilizan en aplicaciones de diseño asistido por ordenador (CAD). Como desventaja presenta que su fidelidad respecto de la imagen real es inferior a la lograda con mapas de bits. Algunos de los principales formatos son: IGES, DxF, EPS, TrueType, etc.

## 2.4. REPRESENTACIÓN DE DATOS NUMÉRICOS

Si se va a realizar alguna operación aritmética, la representación de los datos numéricos como textos es inapropiada.

Ejemplo: Representación y suma de 255 y 42

• Notación en ASCII

255 = (001100100011010100110101)<sub>ASCII</sub>

42 = (0011 0100 0011 0010)<sub>ASCII</sub>

0011 0010 0011 0101 0010 0101  
+                   0011 0100 0011 0010  
-----  
0011 0010 0110 1001 0101 0111

→ 2iW)<sub>ASCII</sub> → ¡MAL!

En este caso, los computadores utilizan representaciones específicas para valores numéricos.

Hay dos formas de representar los datos numéricos:

- Datos de tipo entero.
- Datos de tipo real.

### 2.4.1. NÚMEROS ENTEROS

Para la representación de números enteros se distinguen dos formas de representarlos en el interior del computador: representación binaria y representación BCD, distinguiéndose dentro de cada una de ellas varios métodos distintos de codificación.

#### 1) Representación binaria

- Enteros sin signo (valor absoluto): todos los bits del dato representan el valor del número expresado en binario natural. Con  $n$  bits se pueden representar  $2^n$  números.
- Enteros con signo
  - Signo y magnitud: se reserva el bit más significativo para indicar el signo (0 para positivos y 1 para negativos), y el resto de bits indican la magnitud.
  - Complemento a 1: la representación de un número positivo en complemento a uno es igual que con el sistema signo-magnitud, mientras que la representación de un número negativo consiste en invertir los bits del valor positivo.
  - Complemento a 2: la representación de un número positivo en complemento a dos es igual que con el sistema signo-magnitud, mientras que la representación de un número negativo se calcula obteniendo en primer lugar el complemento a 1 y sumándole 1 al resultado obtenido.
  - Sesgada (o en exceso): todos los valores se representan sumándole un sesgo o exceso. Se suele tomar como sesgo  $S = 2^{n-1}$

#### 2) Representación BCD

Se representan codificando individualmente cada cifra decimal en binario con 4 bits. Existen varias codificaciones posibles:

- BCD natural: los pesos de cada posición son: 8-4-2-1
- BCD Aiken: los pesos de cada posición son: 2-4-2-1
- BCD exceso a 3: se codifica en BCD natural y se suma 3 al resultado.

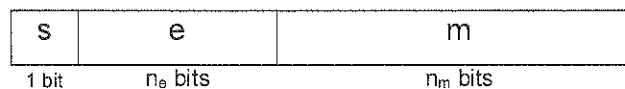


### 2.4.2. NÚMEROS REALES

Para la representación de números reales, se utiliza la notación en coma flotante de la siguiente forma:  $N = M \times B^E$  donde M: mantisa, B: base, E: exponente

Ejemplo:  $N = 1,07653 \times 10^{12}$

En la actualidad, la mayoría de fabricantes de computadores siguen la representación normalizada IEEE754 para la representación de números reales, cuyo esquema es el siguiente:



Donde el significado de cada campo es:

***Campo del signo (s)***

$s=0$ , si  $N > 0$

$s=1$ , si  $N < 0$

***Campo del exponente (e)***

El exponente se almacena en la forma de entero sesgado, donde  $S = 2^{n_e-1}-1$  y

$e = E + S$

***Campo de la mantisa (m)***

Cuando el 1 más significativo de la mantisa se encuentra en la posición de las unidades el número se dice que está normalizado.

El campo de la mantisa solo almacena la parte fraccionaria del número normalizado, puesto que el 1 siempre estará (se dice que está implícito)

Por tanto  $1 \leq M < 2$   $M = 1, m$   
 Ejemplo:  $101,111 \times 2^{-3} = 1,011111 \times 2^{-3}$   $m = 011110...0$

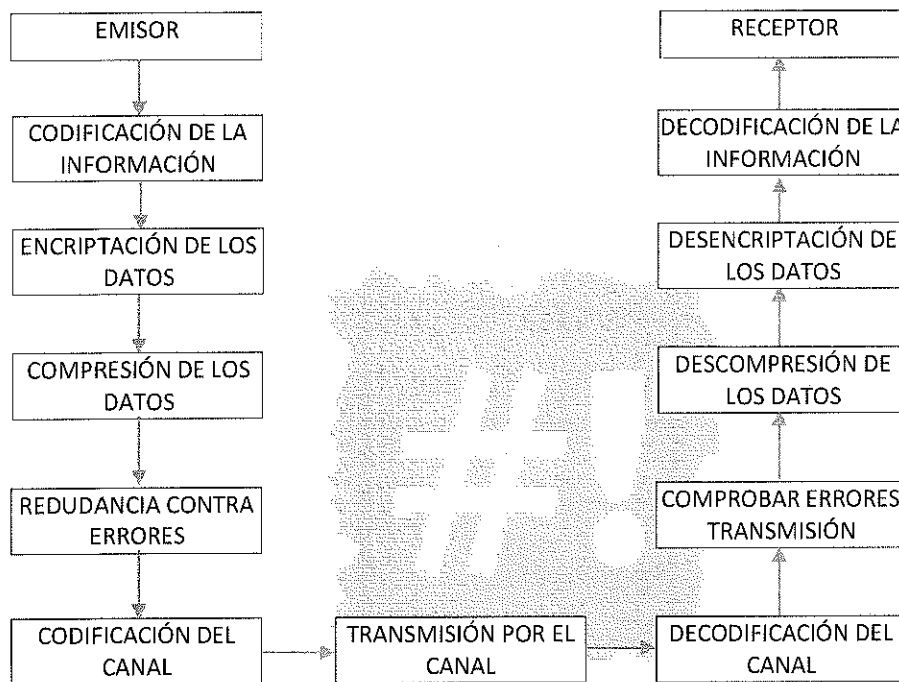
Existen varios formatos de IEEE 754:

- ✓ De 32 bits:
  - 1 bit para el signo, 8 bits para el exponente, 23 bits para la mantisa
- ✓ De 64 bits:
  - 1 bit de signo, 11 bits para el exponente, 52 bits para la mantisa
- ✓ De 128 bits:
  - 1 bit de signo, 15 bits para el exponente, 112 bits para la mantisa

### 3. COMUNICACIÓN DE LA INFORMACIÓN

La comunicación es el proceso mediante el cual un emisor envía información a un receptor a través de un canal.

Centrándonos en un sistema de comunicación digital:



El emisor es el encargado de generar la información a transmitir. Dependiendo del tipo de información, esta se codifica mediante la transformación más adecuada (la codificación de la información se ha visto en el punto anterior de representación de la información). Opcionalmente se pueden aplicar una serie de alteraciones al mensaje:

- Puede ser **cifrado** para evitar que terceros no autorizados conozcan su contenido.
- Puede ser **comprimido** para que el tamaño de la información transmitida sea menor.
- Puede incorporar **redundancias** para mitigar los errores producidos durante la transmisión.

Y finalmente será **codificado** en una señal para adaptarlo al medio por el que debe viajar.

### 3.1. ENCRIPCIÓN DE DATOS

La encriptación o cifrado transforma una información (texto claro) en otra ininteligible (texto cifrado), con el objetivo de garantizar su confidencialidad.

Básicamente se distinguen dos sistemas de cifrado:

- **Cifrado simétrico:** es aquel en el que las claves para cifrar y descifrar son iguales. La seguridad del proceso depende del secreto de la clave. Ejemplos: DES, TDES, IDEA, AES, Blowfish, RC4, RC5, RC6, SEAL, SAFER.
- **Cifrado asimétrico:** es aquel en que las claves para cifrar y descifrar son distintas (clave pública y clave privada). Ejemplos: RSA, DSA, Diffie-Hellman.

### 3.2. COMPRESIÓN DE DATOS

La información puede ser comprimida con el fin de reducir su tiempo de transmisión.

Básicamente se distinguen dos sistemas de compresión:

- **Compresión sin pérdidas:** la información se comprime sin perder nada de la información inicial. Ejemplos de técnicas de compresión son: RLE, Lempel-Ziv LZ77, etc.
- **Compresión con pérdidas:** para conseguir una mayor compresión se toleran pérdidas de calidad, no pudiéndose en la descompresión recuperar exactamente la información de partida. Ejemplos de técnicas de compresión son: GIF, MPEG, MP3, etc.

### 3.3. REDUNDANCIA DE DATOS

Para detectar y corregir la perturbación producida en la transmisión de la información se utilizan determinados códigos que añaden una secuencia de bits (redundancia) al mensaje.

Básicamente estos códigos se pueden agrupar en:

#### a) Códigos detectores de errores:

- **Código de paridad:** permite detectar errores de un bit. Existen dos criterios:
  - Paridad par: se añade un bit (0 o 1) de forma que el número total de unos del código sea par.
  - Paridad impar: se añade un bit (0 o 1) de forma que el número total de unos del código sea impar.
- **Código 2 entre 5:** emplea 5 dígitos para representar cada número, de forma que el primero es un bit de paridad y los otros cuatro tienen los pesos 1, 2, 3 y 6, y solamente puede haber dos bits a 1 entre los cinco bits.
- Etc.

#### b) Códigos correctores de errores:

- **Código Hamming:** Permite detectar y corregir errores. Incorpora bits de control en las posiciones potencia de dos.
- **Códigos de redundancia cíclica (CRC):** el emisor y el receptor comparten un número (polinomio generador) entre el que dividen todo el grupo de bits a enviar. El emisor añade el módulo obtenido en la división al mensaje como bits de redundancia, de forma que debe ser divisible por el polinomio generador.

#### 4. RECURSOS Y HERRAMIENTAS EDUCATIVAS DE INTERÉS

– **IEEE-754 Analysis**

Es una aplicación web desarrollada por el Queens College que permite examinar las relaciones entre los números binarios y decimales y los tres formatos de números descritos por el estándar de punto flotante IEEE-754-2008. Sitio web: <https://babbage.cs.qc.cuny.edu/ieee-754/>

– **HexEd.it**

Es una aplicación web que permite subir archivos de texto plano guardados con diferentes formatos y muestra la codificación empleada para almacenar cada carácter. Sitio web: <https://hexed.it>

#### 5. CONCLUSIÓN

En este tema se ha presentado una visión global de los aspectos fundamentales relacionados con la representación de la información en un computador, ya que son clave para poder diseñar sus componentes estructurales.

En la segunda mitad del tema se han introducido conceptos relacionados con la comunicación y transmisión de la información como son: la encriptación y compresión de datos, así como técnicas para la redundancia contra errores producidos durante la transmisión.

#### 6. BIBLIOGRAFÍA

- Prieto A., y otros. *Introducción a la informática*. Editorial McGraw-Hill
- De Miguel Anasagasti, P. *Fundamentos de los computadores*. Editorial Paraninfo.
- García de Sola, J. F. *Estructura de la información*. Editorial McGraw-Hill
- [http://atc.ugr.es/APrieto\\_videoclases](http://atc.ugr.es/APrieto_videoclases) Departamento de Arquitectura y Tecnología de Computadores. Universidad de Granada.
- [www.unicode.org](http://www.unicode.org) (Consortio Unicode)
- <https://hexed.it>
- <https://babbage.cs.qc.cuny.edu/ieee-754/>