

Unidades de Información

16 de marzo de 2017

Índice

Unidades de Información	1
Información	1
Bit	2
El viaje de un bit	3
Byte	3
Representando datos con bytes	4
Sistema Internacional	4
Sistema de Prefijos Binarios	4

Unidades de Información

En este segundo tema de la unidad veremos qué es la información y cómo podemos cuantificar, es decir, medir, la cantidad de información que puede alojar un dispositivo, o la cantidad de información que representa una pieza cualquiera de información. Veremos además las relaciones entre las diferentes unidades de información.

Información

A lo largo de la historia se han inventado y fabricado máquinas, que son dispositivos que **transforman la energía**, es decir, convierten una forma de energía en otra. Las computadoras, en cambio, convierten una forma de **información** en otra.

Los programas de computadora reciben alguna forma de información (la **entrada** del programa), la **procesan** de alguna manera, y emiten alguna información de **salida**. La **entrada** es un conjunto de datos de partida, para que trabaje el programa, y

la **salida** generada por el programa es alguna forma de respuesta o solución a un problema. Sabemos, además, que el material con el cual trabajan las computadoras son textos, documentos, mensajes, imágenes, sonido, etc. Todas estas son formas en las que se codifica y se almacena la información.

Un epistemólogo dice que la información es “una diferencia relevante”. Si vemos que el semáforo cambia de rojo a verde, recibimos información (“podemos avanzar”). Al cambiar el estado del semáforo aparece una **diferencia** que puedo observar. Es **relevante** porque modifica de alguna forma el estado de mi conocimiento o me permite tomar una decisión respecto de algo.

¿Qué es, exactamente, esta información? No podemos tocarla ni pesarla, pero ¿se puede medir? Y si se puede medir, ¿entonces se puede medir la cantidad de información que aporta un texto, una imagen?

Bit

La Teoría de la Información, una teoría matemática desarrollada alrededor de 1950, dice que el **bit** es “la mínima unidad de información”. Un bit es la información que recibimos “cuando se especifica una de dos alternativas igualmente probables”. Si tenemos una pregunta **binaria**, es decir, aquella que puede ser respondida **con un sí o con un no**, entonces, al recibir una respuesta, estamos recibiendo un bit de información. Las preguntas binarias son las más simples posibles (porque no podemos decidir entre **menos** respuestas), de ahí que la información necesaria para responderlas sea la mínima unidad de información.

De manera que un bit es una unidad de información que puede tomar sólo dos valores. Podemos pensar estos valores como **verdadero o falso**, como **sí o no**, o como **0 y 1**.

Cuando las computadoras trabajan con piezas de información complejas, como los textos o imágenes, estas piezas son representadas como conjuntos ordenados de bits, de un cierto tamaño. Así, por ejemplo, la secuencia de ocho bits **01000001** puede representar la letra A mayúscula. Un documento estará constituido por palabras; éstas están formadas por símbolos como las letras, y éstas serán representadas por secuencias de bits.

La memoria de las computadoras está diseñada de forma que **no se puede almacenar otra cosa que bits** en esa memoria. Los textos, las imágenes, los sonidos, los videos, los programas que ejecuta, los mensajes que recibe o envía; todo lo que puede guardar, procesar, o emitir una computadora digital, debe estar en algún momento representado por una secuencia de bits. Los bits son, en cierta forma, como los átomos de la información. Por eso el bit es la unidad fundamental que usamos para medirla, y definiremos también algunas unidades mayores, o múltiplos.

El viaje de un bit

En una famosa película de aventuras hay una ciudad en problemas. Uno de los héroes enciende una pila de leña porque se prepara un terrible ataque sobre la ciudad. La pila de leña es el dispositivo preestablecido que tiene la ciudad para pedir ayuda en caso de emergencia.

En la cima de la montaña que está cruzando el valle existe un puesto similar, con su propio montón de leña, y un vigía. El vigía ve el fuego encendido en la ciudad que pide ayuda, y a su vez enciende su señal. Lo mismo se repite de cumbre en cumbre, atravesando grandes distancias en muy poco tiempo, hasta llegar rápidamente a quienes están en condiciones de prestar la ayuda. En una tragedia griega se dice que este ingenioso dispositivo se utilizó en la realidad, para comunicar en tan sólo una noche la noticia de la caída de Troya.

La información que está transportando la señal que viaja es la respuesta a una pregunta muy sencilla: “¿la ciudad necesita nuestra ayuda?”.

Esta pregunta es **binaria**: se responde con un sí o con un no. Por lo tanto, lo que ha viajado es **un bit de información**.

Notemos que, en los manuales de lógica o de informática, encontraremos siempre asociados los **bits** con los valores de **0 y 1**. Aunque esto es útil a los efectos de emplear los bits en computación, no es del todo exacto. Lo que está codificando un bit no es exactamente un dígito. Lo que viajó desde la ciudad sitiada hasta su destino no es un 0 ni un 1. Es **un bit de información**, aquella unidad de información que permite tomar una decisión entre dos alternativas.

Byte

Como el bit es una medida tan pequeña de información, resulta necesario definir unidades más grandes. En particular, y debido a la forma como se organiza la memoria de las computadoras, es útil tener como unidad al **byte** (abreviado **B** mayúscula), que es una secuencia de **8 bits**. Podemos imaginarnos la memoria de las computadoras como una estantería muy alta, compuesta por estantes que contienen ocho casilleros. Cada uno de estos estantes es una **posición o celda de memoria**, y contiene exactamente ocho bits (un byte) de información.

Como los valores de los bits que forman un byte son independientes entre sí, existen 2^8 diferentes valores para esos ocho bits. Si los asociamos con números en el sistema binario, esos valores serán **00000000, 00000001, 00000010, ...**, etc., hasta el **11111111**. En decimal, esos valores corresponden a los números **0, 1, 2, ...**, **255**.

Cada byte de la memoria de una computadora, entonces, puede alojar un número entre 0 y 255. Esos números representarán diferentes piezas de información: si los vemos como bytes independientes, pueden representar **caracteres** como letras y otros símbolos, pero también pueden estar formando parte de otras estructuras de información más complejas, y tener otros significados.

Representando datos con bytes

Para poder tratar y comunicar la información, que está organizada en bytes, es necesario que exista una asignación fija de valores binarios a caracteres. Es decir, se necesita una **tabla de caracteres** que asigne un símbolo a cada valor posible entre 0 y 255.

La memoria de la computadora es como un espacio donde se almacenan temporariamente contenidos del tamaño de un byte. Si pudiéramos ver el contenido de una sección de la memoria mientras la computadora está trabajando, veríamos que cada byte tiene determinados contenidos binarios. Esos contenidos pueden codificar los caracteres de un mensaje, carácter por carácter.

Sabiendo que la memoria está organizada en bytes, es interesante saber qué capacidad tendrá la memoria de una computadora dada y qué tamaño tendrán las piezas de información que caben en esta memoria. Como la memoria de una computadora, y la cantidad de información que compone un mensaje, un programa, una imagen, etc., suelen ser muy grandes, utilizamos **múltiplos** de la unidad de memoria, que es el byte.

Existen en realidad dos sistemas diferentes de múltiplos: el **Sistema Internacional** y el **Sistema de Prefijos Binarios**. Las unidades de ambos sistemas son parecidas, pero no exactamente iguales.

Los dos sistemas difieren esencialmente en el factor de la unidad en los sucesivos múltiplos. En el caso del Sistema Internacional, todos los factores son alguna potencia de 1000. En el caso del Sistema de Prefijos Binarios, todos los factores son potencias de 1024.

Sistema Internacional

En el llamado Sistema Internacional, la unidad básica, el byte, se multiplica por potencias de 1000. Así, tenemos:

- El **kilobyte** (1000 bytes)
- El **megabyte** (1000×1000 bytes = 1000 kilobytes = un millón de bytes)
- El **gigabyte** ($1000 \times 1000 \times 1000$ bytes = mil megabytes = mil millones de bytes)
- El **terabyte** ($1000 \times 1000 \times 1000 \times 1000$ bytes = mil gigabytes = un billón de bytes), y otros múltiplos mayores.

Como puede verse, cada unidad se forma multiplicando la anterior por 1000.

Sistema de Prefijos Binarios

En el llamado Sistema de Prefijos Binarios, el byte se multiplica por potencias de 2^{10} , que es 1024. Así, tenemos:

- El **kibibyte** (1024 bytes)
- El **mebibyte** (1024×1024 bytes, **aproximadamente** un millón de bytes, pero exactamente 1048576 bytes)
- El **gibibyte** ($1024 \times 1024 \times 1024$ bytes, **aproximadamente** mil millones de bytes)
- El **tebibyte** ($1024 \times 1024 \times 1024 \times 1024$ bytes, aproximadamente un millón de mebibytes, o aproximadamente un billón de bytes), y otros múltiplos mayores.

Como puede verse, cada unidad se forma multiplicando la anterior por 1024 .

Notemos que los prefijos **kilo**, **mega**, **giga**, **tera**, del Sistema Internacional, cambian a **kibi**, **mebi**, **gibi**, **tebi**, del sistema de Prefijos Binarios.

¿Por qué existen **dos** sistemas en lugar de uno? En realidad la adopción del Sistema de Prefijos Binarios se debe a las características de la memoria de las computadoras:

- Cada posición o celda de la memoria tiene su dirección, que es el número de la posición de esa celda dentro del conjunto de toda la memoria de la computadora.
- Cuando la computadora accede a una posición o celda de su memoria, para leer o escribir un contenido en esa posición, debe especificar la dirección de la celda.
- Como la computadora usa exclusivamente números binarios, al especificar la dirección de la celda usa una cantidad de dígitos binarios.
- Por lo tanto, la cantidad de posiciones que puede acceder usando direcciones es una potencia de 2 : si usa 8 bits para especificar cada dirección, accederá a 2^8 bytes, cuyas direcciones estarán entre 0 y 255 . Si usa 10 bits, accederá a 2^{10} bytes, cuyas direcciones serán 0 a 1023 .
- Entonces, tener una memoria de, por ejemplo, exactamente **mil bytes**, complicaría técnicamente las cosas porque las direcciones 1000 a 1023 no existirían. Si un programa quisiera acceder a la posición 1020 habría un grave problema. Habría que tener en cuenta excepciones por todos lados y la vida de los diseñadores de computadoras y de los programadores sería lamentable.
- En consecuencia, todas las memorias se fabrican en tamaños que son potencias de 2 y el Sistema de Prefijos Binarios se adapta perfectamente a medir esos tamaños.

En computación se utilizan, en diferentes situaciones, ambos sistemas de unidades. Es costumbre usar el Sistema Internacional para hablar de velocidades de transmisión de datos, pero usar Prefijos Binarios al hablar de almacenamiento de memoria, o en unidades de almacenamiento permanente, como los discos.

- Entonces, cuando un proveedor de servicios de Internet ofrece **un enlace de 1 Mbps**, nos está diciendo que por ese enlace podremos transferir **exactamente 1 millón de bits por segundo**. El proveedor utiliza el Sistema Internacional.

- Los textos, imágenes, sonido, video, programas, etc., se guardan en **archivos**, que son sucesiones de bytes. Encontramos archivos en el disco de nuestra computadora, y podemos descargar archivos desde las redes. Cuando nos interesa saber cuánto mide un archivo, en términos de bytes, usamos el Sistema Internacional porque el archivo no tiene por qué tener un tamaño que sea potencia de 2.
- Por el contrario, los fabricantes de medios de almacenamiento, como memorias, discos rígidos o pendrives, deberían (aunque a veces no lo hacen) utilizar Prefijos Binarios para expresar las capacidades de almacenamiento de esos medios. Así, un “*pendrive de dieciséis gigabytes*”, si tiene una capacidad de 16×2^{30} bytes, debería publicitarse en realidad como “*pendrive de dieciséis gibibytes*”.