



INFORMÁTICA

décimo grado

Informática

Décimo grado

M. Sc. Ilien Pérez González
M. Sc. César Labañino Rizzo
Lic. Tatiana Kindelán Duharte



Colaborador:

- Dr. C. Yunior Portilla Rodríguez

Edición y corrección:

- Lic. Yaimara Borges Forcades

Diseño de cubierta:

- Instituto Superior de Diseño • Daniela Alpízar Cespedes • Laura Rosa Almero Fong

Diseño e ilustración:

- Instituto Superior de Diseño

Instituto Superior de Diseño:

Anelís Simón Sosa • María Paula Lista Jorge • Amanda Serrano Hernández • Ana Laura Seco Abreu • Arianna Ruenes Torres • Celia Carolina Céspedes Pupo • Elizabeth Diana Fajardo Céspedes • Laura Rosa Almero Fong • Elizabeth Blanco Galbán • Daniela Arteaga Martínez • Daniela Alpízar Cespedes • Camila Noa Clavero • Daniel Fabelos Rodríguez • M. Sc. Maité Fundora Iglesias • Dr. C. Ernesto Fernández Sánchez • D. I. Eric Cuesta Machado • D. I. Julio Montesino Carmona

Emplane:

- Elizabeth Blanco Galban

© Ilien Pérez González y coautores, Cuba, 2023

© Editorial Pueblo y Educación, 2023

ISBN 978-959-13-4168-6 (Versión impresa)

ISBN 978-959-13-4360-4 (Versión digital)

EDITORIAL PUEBLO Y EDUCACIÓN

Ave. 3. º A No. 4601 entre 46 y 60,

Playa, La Habana, Cuba. CP 11300.

epe@enet.cu



ÍNDICE

Prólogo	V
----------------------	----------

1

Recursos multimedia	1
1.1 Tratamiento y principios de diseño de la imagen digital estática ..1	
1.1.1 Introducción a las imágenes digitales	2
1.1.2 Características de las imágenes de mapa de bits	9
1.1.3 Los formatos de la imagen digital	21
1.1.4 La composición, el retoque y la optimización de la imagen digital .	26
1.1.5 Los sistemas para el trabajo de la imagen digital	27
1.1.6 La edición, el fotomontaje y el retoque de imágenes digitales .	31
1.2 Tratamiento del sonido digital	86
1.2.1 Características del sonido digital	89
1.2.2 Componentes del hardware, necesarios para manejar el sonido digital	98
1.2.3 Procesamiento digital del sonido	101
1.3 Tratamiento y edición de video	116
1.3.1 Edición del video digital	120

2

Introducción a la programación orientada a objetos y conducida por eventos	133
2.1 El Pensamiento computacional	134
2.1.1 Definición de problema	138
2.2 Datos y expresiones	148
2.2.1 Los datos	149
2.2.2 Expresiones	155
2.2.3 Contadores y acumuladores	157
2.2.4 Descripción de algoritmos	159
2.2.5 Elementos de programación	167

2.3	La programación orientada a objetos y dirigida por eventos	173
2.3.1	Breve reseña histórica de LiveCode, algunas de sus características	176
2.3.2	Los eventos, los objetos y sus propiedades en LiveCode	182
2.3.3	Propiedades personalizadas o propiedades definidas por el usuario	186
2.3.4	Entrada y salida	187
Bibliografía		201

Prólogo

Estimado educando, ponemos en tus manos este libro de texto de Informática con el objetivo de que poseas un medio de enseñanza que aborde los contenidos del grado en una asignatura que, como sabes, es de suma importancia, pues los grandes adelantos en equipos de computación, sistemas de información y telecomunicaciones han transformado las economías mundiales en economías basadas en el conocimiento, las comunicaciones y la información, y es necesario que estés capacitado para contribuir directamente tanto en el desarrollo de nuestro país, como en el tuyo personal como ciudadano, con tus innovaciones, mediante el empleo de las tecnologías de la información.

En este libro, quisiéramos compartir con todos ustedes, jóvenes que inician sus estudios de la Educación Media Superior, el gusto por la computación, así como los conocimientos para utilizar las computadoras como herramientas valiosas en sus quehaceres y en su futura vida profesional. Es por ello necesario que conozcas los conceptos generales y cómo utilizarlas de manera eficiente. El trabajar con ellas representa un reto y es un ejercicio estimulante para el intelecto, ya que las actividades y problemas propuestos te brindan la posibilidad de desarrollar aspectos prácticos para su utilización.

El libro contiene dos capítulos: “Recursos multimedia” e “Introducción a la Programación orientada a objetos y dirigida por eventos”. En el capítulo 1 se aborda, a partir de tres temáticas fundamentales, la creación y edición de imágenes digitales, el tratamiento del sonido digital y la edición de videos.

En el capítulo 2 también se trabaja en tres temas, el primero aborda lo referente a un contenido muy reciente en el mundo de la informática, se trata de lo que se conoce hoy en día como Pensamiento computacional, que no es más que el sistema de conocimientos, habilidades y actitu-

des asociadas con la solución de problemas por vías informáticas. Este ***pensamiento***, se concibe hoy como los fundamentos básicos acerca del funcionamiento de los sistemas informáticos y es nuestra intención que des pasos en esa dirección para que dejes de ser un consumidor y te conviertas en un creador de productos informáticos, para lo cual resulta necesario adoptar formas de pensamiento específicas, que modernamente han devenido en llamarse Pensamiento computacional. En este capítulo se trabajan invariantes conceptuales de esta temática, lo que te va a preparar para el estudio posterior de cualquier lenguaje de programación.

En el segundo tema conocerás los elementos básicos de la información, o sea, los datos y formas de procesamiento mediante expresiones, y por último, el tercer tema te introduce en la Programación orientada a objetos (POO), en el caso particular de una programación basada en objetos y dirigida por eventos. Para cerrar este capítulo se propone la implementación de algoritmos mediante un lenguaje de programación moderno y versátil.

Esperamos que hagas un uso óptimo de este medio de enseñanza y te sirva de estímulo para que des pasos firmes en este apasionante mundo de la Informática.

Los autores de este libro se lo dedicamos a todas las muchachas que cursan el preuniversitario y en especial a nuestras hijas, Indira, Rocío, Rosalí, Nathalie y Camila para que, al igual que Ada Byron, rompan el estigma de que las mujeres no pueden ser programadoras.

CAPÍTULO 1

Recursos multimedia

Los modernos medios de comunicación —prensa, televisión, cine, radio, etc.— han contribuido en gran medida a crear un entorno social cada vez más amplio. Por la influencia que ellos ejercen, muchas personas en todo el mundo comparten gustos, aficiones y opiniones, por encima de etnias, lenguajes y países.

En dichos medios de comunicación, la imagen desempeña un gran papel por la fuerza que tiene el mensaje que trasmite y por su facilidad de comprensión; basta con observar las diversas señales que proliferan en los diferentes lugares públicos, los iconos que aparecen en la pantalla de nuestra computadora o los símbolos que sustituyen a las palabras en etiquetas y aparatos. Por consiguiente, la comunicación y el lenguaje de la imagen forman un binomio inseparable.

1.1 Tratamiento y principios de diseño de la imagen digital estática

La imagen digital, por su naturaleza económica, es de fácil duplicación, ya que no gasta materiales en su producción y no hay que crear nuevos recursos de **hardware** para trabajarla, excepto los que generalmente poseemos.

Su gran valor está en los múltiples usos que tiene en el mundo contemporáneo, desde el punto de vista docente está dado por la capacidad de resaltar determinados detalles o enriquecerlos por parte del profesor, lo que hace que su estudio sea de vital importancia en todas aquellas temáticas que abordan la informática educativa. A su vez, la gran variedad de transformaciones que puede sufrir este tipo de imagen ha de mantenernos siempre alertas, pues de hecho la realidad puede ser cambiada de

forma malintencionada, ya que se vive en un mundo donde el acceso a la tecnología y la información es cada día mayor, al disponerse de la Internet y otras redes donde la esfera comercial tiene un gran control y el diversio-nismo ideológico está cada vez más sobre nuestras espaldas.

En este primer epígrafe comenzarás el estudio de la imagen digital y en particular, la creación y modificación de imágenes fijas. En una primera parte, además de una breve presentación de los componentes generales sobre la imagen fija, encontrarás una introducción al dibujo vectorial, utilizando como herramienta básica de trabajo el programa Draw, del ya conocido paquete de aplicaciones ofimáticas LibreOffice. A continua-ción, se abordarán los conceptos sobre dibujo en bitmap o mapa de bits, el retoque fotográfico y el fotomontaje; usando como herramientas las siguientes aplicaciones: GIMP, uno de los más populares programas de re-toques fotográfico de distribución libre, gratuita y multiplataforma, y el ya conocido Photoshop de Adobe.

1.1.1 Introducción a las imágenes digitales

“Una imagen vale más que mil palabras”. ¿Estás de acuerdo con esta afirmación? ¿Será esto cierto? Al observar una imagen se pue-de decir, sin duda alguna, que es más “gráfica”, que puede captar la esencia de una situación, pero, ¿qué nos dice la foto de la figura 1.1?, ¿a qué hace referencia? La foto capta un momento, pero refleja un con-texto; aunque para conocer ese contexto se necesita un poco más de información, además de la ima-gen, quizás un texto.



Fig. 1.1

Con esto, habrá que modificar esa frase tan usada que nos dice que una imagen vale más que mil palabras; más bien ella las resume, las congela en un momento. La forma de percibir al observar una imagen es distinta, no se procesa la información por la razón, sino por la emoción. La imagen es, por tanto, más poderosa que el texto y puede captar nuestra atención

en un segundo, pero si se quiere conocer lo que ha pasado, entonces se necesita el contexto.

El ser humano es eminentemente visual. Hoy en día, no se entiende bien la información sin una imagen que congele o que nos dé un detalle (intencionado o no) de un hecho, y las modernas tecnologías de la información y las comunicaciones (TIC) permiten que cualquiera de los usuarios de la tecnología digital tenga en sus manos, la posibilidad de colocarse en la silla del creador, de asumirse como un generador de contenido y de comunicar, por medio de imágenes digitales, sus ideas; con la convicción de ser un usuario colaborativo o más participativo en la esfera social-digital.

¿Qué es una imagen digital?

Actualmente, se vive en una era en la que todas las formas de la información están sufriendo un proceso de digitalización. Las imágenes, como ya has de saber, no han podido escapar a este proceso. La fotografía, el cine, la televisión, el diseño gráfico, e incluso el diseño industrial, producen miles de imágenes digitales que son almacenadas en algún soporte físico, presentadas en una pantalla e impresas en papel o en algún dispositivo de almacenamiento, o enviadas por un medio de transmisión electrónico.



Algo de historia

En un inicio, el ser humano le dio a la imagen un tratamiento pictórico, en el que mediante el tallado, grabado o pintura en piedra o sobre diversos lienzos, logró plasmar lo que en ese entonces era capaz de representar. Con la imprenta, la masificación de la imagen cobró nuevas dimensiones y con la llegada de las técnicas fotográficas para su captura, se alcanzó la cúspide en el campo de la generación de imágenes en formato analógico. Hoy en día, además de los métodos anteriormente mencionados para la generación de imágenes, es posible generar fotografías, ilustraciones e imágenes mediante el **hardware** y el **software** adecuados.

La generación de imágenes digitales es una actividad altamente recurrente. La existencia de diversos tipos de dispositivos para la toma de fotografías, y la sencillez de realizar esta tarea, han propiciado el incremento de la generación y utilización de fotografías digitales. Además, son rápidas, económicas y se pueden alojar en la red, por lo que su difusión

deja de depender del creador, ya que los usuarios van trazando el camino que seguirán al encontrarlas, duplicarlas, modificarlas y retransmitirlas.



Definición

Una **imagen digital** es básicamente un tipo de imagen que se obtiene usando dispositivos electrónicos (escáner, cámara digital), y puede ser creada y manipulada mediante un sistema informático.

La información digital que genera cualquiera de los dispositivos de conversión analógico-digital como un escáner, una cámara fotográfica digital, es almacenada en la computadora mediante **bits**.



Recuerda que...

Un **bit** es la unidad de información que equivale a la elección de dos posibilidades: 0 o 1 (no pasa corriente o pasa corriente). Los circuitos internos del sistema informático (computadoras, tabletas, teléfonos inteligentes) solo detectan estos dos valores y en la combinación de ellos se basa la informática. El nombre proviene de la contracción de las palabras **binary** y **digit** (dígito binario).

La información que se puede almacenar en un bit es muy pequeña, por eso los bits se agrupan en **bytes**, que es un grupo de 8 bits (cada uno de ellos puede representar un valor diferente: 0 o 1). De esta forma, un byte puede representar cualquier número entre 0 y 255. A continuación, se representan las unidades de medida de la información digital:

- Byte **Byte** (B) 8 bits
- Kilobyte **Kbyte** (Kb) 1 024 bytes
- Megabyte **Mbyte** (Mb) 1 024 Kbytes
- Gigabyte **Gbyte** (Gb) 1 024 Mbytes
- Terabyte **Tbyte** (Tb)
- Petabyte **Pbyte** (Pb)

Cómo puedes ver las imágenes digitales están formadas por una serie de matrices numéricas de ceros y unos que se almacenan en una memoria informática, y que definen las características de una fotografía. Una vez que esta imagen es interpretada por la computadora, se transforma en

una imagen visible a través de la pantalla e imprimible también por medio de cualquier dispositivo de salida (figura 1.2).



Esta imagen

```
101010100101010101010100100100101110110
00111111101010010010010101010100110101001
0100101001010010100101001010010101010101
010100101001010100101001010101001010101
00101010010100101001010100101010001001001
0110010101001010101001011100101011101001
01010010111010010100010011111001010101011
01110010101010101000101010111001001001011
00101010101010010010100101010010101001010
100101001010010101010101010010101010010
101010010101010100101010101001010100101
01001010101010101010100101010101010010
```

Se almacena de la siguiente forma

Fig. 1.2

Si hablamos de ventajas, la imagen digital tiene varias, como son:

- Facilidad para distribuirse en las redes.
- Posibilidad de añadir datos extras.
- Copias sin pérdida de calidad.
- Mayor duración del original.
- Facilidad de almacenaje.
- Color de mayor calidad y más modificable.

Pero, la gran ventaja del archivo digital es que puede duplicarse y copiarse tantas veces como se quiera.



Algo de historia

La primera imagen digital de la historia no fue tomada por la primera cámara digital que se inventó en 1975. Es que esta primera imagen digital se había obtenido ya dos décadas antes. Bien, lo que sucedió fue que en 1957 Russell Kirsch creó una imagen digital de 176×176 píxel, empleando un dispositivo que transformaba las imágenes en matrices de ceros y unos. Aquel photo-escáner tomó como primer modelo una imagen del bebé de tres meses de edad de su operador. La bajísima resolución de la imagen —que pueden ver junto a estas líneas— era inevitable teniendo en cuenta la casi nula capacidad de almacenamiento que tenía el ordenador al que iba unido aquel primitivo escáner. Poco podía sospechar el pequeño bebé Kirsch, que su rostro entraría de lleno en la historia de la tecnología. ¡Verdad!



La presencia o no de movimiento en las imágenes digitales permite clasificarlas ante todo en dos grandes grupos, que difieren en cuanto a formato y tratamiento: la **imagen estática** y la **imagen dinámica**. En este capítulo solo se va a realizar el estudio de las imágenes digitales estáticas.

Las **imágenes digitales estáticas** se dividen en dos grandes tipos: **imágenes vectoriales** y las de **mapa de bits**. Dependiendo de esta clasificación, la imagen se deberá modificar con un tipo de herramienta determinada y se deberá emplear para aquellos fines donde la imagen elegida funcione mejor. Esta no es una división categórica, ya que las imágenes vectoriales suelen admitir la incrustación de imágenes de mapa de bits en su interior, y los programas especializados en dibujo vectorial (Adobe Illustrator, Macromedia Freehand y Corel Draw), cada vez tienen más cualidades de los programas de tratamiento de imágenes de mapa de bits (Adobe Photoshop o GIMP).

Debes saber que existen otros tipos de imágenes digitales, como son las llamadas 3D, que transmiten la proyección de un objeto visual en un espacio tridimensional. Las **imágenes 3D** son muy utilizadas en aplicaciones de arquitectura y diseño industrial, ya que permiten generar representaciones fieles de objetos reales desde cualquier punto de vista.



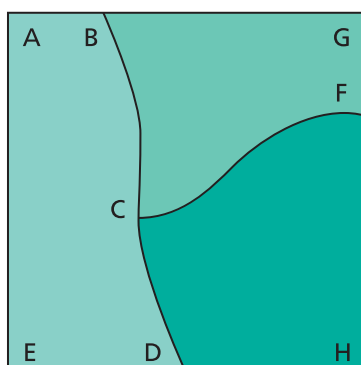
Saber más

Las imágenes también se pueden clasificar como **icónicas** (de diferentes grados de figuración y realismo) o **aicónicas** (abstractas, esquemáticas), **tri-dimensionales** y **planas**, y **fijas** o **móviles**.

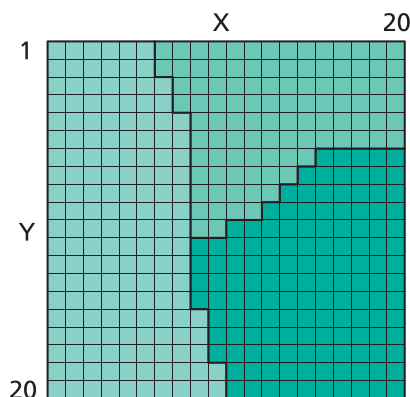
¿A qué se denomina imagen de mapa de bits? Una imagen en **mapa de bits**, **imagen rásters** (calcos del inglés) o **imagen de píxeles**, es una estructura o fichero de datos que representa una rejilla rectangular de píxeles o puntos de color, denominada matriz, que se puede visualizar en un monitor, papel u otro dispositivo de representación.

Los gráficos en mapa de bits se distinguen de los gráficos vectoriales en que los ráster almacenan simplemente el color de cada punto en la matriz (figura 1.3 b), mientras que los vectoriales representan una imagen a través del uso de objetos geométricos como curvas (conocidas como curvas de Bezier) y polígonos (figura 1.3 a). El formato de imagen matricial está ampliamente extendido y es el que se suele emplear para tomar fotogra-

fías digitales y realizar capturas de video. Para su obtención se usan dispositivos de conversión analógica-digital, tales como escáneres y cámaras digitales.



a. Imagen vectorial.



b. Imagen de mapa de bits.

Fig. 1.3

¿Qué es una imagen vectorial? Una **imagen vectorial** es una imagen digital formada por objetos geométricos independientes (segmentos, polígonos, arcos, etc.), cada uno de ellos definido por distintos atributos matemáticos de forma, de posición, de color, etcétera.

Son formas determinadas que se almacenan con muy poco espacio en un archivo, que describe su ubicación, color y textura, sin depender de la resolución. El interés principal de los gráficos vectoriales es que pueden ampliar el tamaño de una imagen tanto como se quiera, sin perder calidad; sin embargo, en el caso de las imágenes de mapa de bits sufren la pérdida de calidad. De la misma forma, permiten mover, estirar y retorcer imágenes de una manera relativamente sencilla. Su uso también está muy extendido en la generación de imágenes en tres dimensiones, tanto dinámicas como estáticas.

Las imágenes de mapa de bits pueden convertirse en imágenes vectoriales de aspecto similar (aunque no totalmente idéntico). Esa transformación se lleva a cabo por medio de un proceso denominado "vectorización", donde se reconocen los patrones geométricos en la imagen de mapa de bits y se generan figuras geométricas a partir de ellos.

Por lo general, para el retoque fotográfico o dibujo libre se utilizan los programas de pintura como GIMP o Photoshop, para el diseño gráfico se consideran más apropiados los programas de dibujo de tipo vectorial, como Draw. No obstante, la mayoría de los programas de diseño gráfico

modernos ofrecen también la posibilidad de insertar objetos individuales de mapa de bits, además de las figuras geométricas de tipo vectorial. De esta forma, se logran combinar las ventajas de ambos tipos de imágenes y se multiplican las posibilidades creativas. Incluso, es posible crear una imagen de mapa de bits con otro programa e insertarlo como un objeto más dentro de un gráfico vectorial.

Todas las computadoras actuales traducen los gráficos vectoriales a mapas de bits, para poder representarlos en pantalla, pues esta está constituida físicamente por píxeles. Por ejemplo, aquí puedes apreciar claramente las diferencias entre las imágenes vectoriales y las de mapa de bits, al ser ampliadas (figura 1.4 a, b).

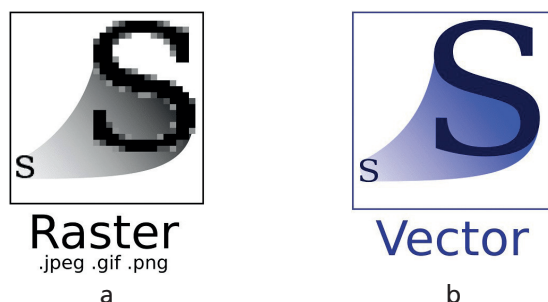


Fig. 1.4

Así pues, antes de crear una imagen hay que elegir una de las dos tecnologías de imagen digital: vectorial o bitmap. Cada una de ellas tiene sus aplicaciones, y su producción exige de unos requisitos que hay que conocer. Las **imágenes vectoriales** son ideales para cartelería, diseño de envases, imagen corporativa, logotipos, etc.; es decir, para todas aquellas situaciones en las que una misma imagen, hecha con una gama reducida de tintas planas, debe ser reproducida en distintos soportes y a distintos tamaños. Las **imágenes en mapa de bits**, en cambio son perfectas cuando la gama de colores cambia sutilmente. En este caso, la imagen debe generarse teniendo muy en cuenta dónde y cómo va a mostrarse, con una cantidad de píxeles y una gama de colores adaptados al soporte en el que va a reproducirse. Una vez hecha, las modificaciones comportarán pérdida de calidad.

A partir de ahora, se hará referencia a la tecnología bitmap, que es la más exigente en lo que concierne a la resolución. Hay que tener pre-

sente este concepto en todas las etapas de la vida de la imagen, desde su captura con una cámara digital o un escáner, hasta su reproducción en el soporte elegido.

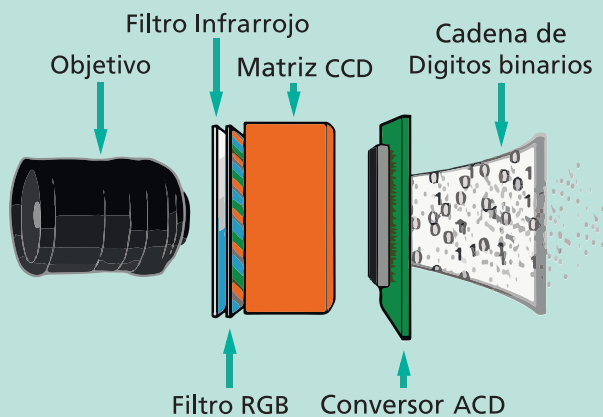
1.1.2 Características de las imágenes de mapa de bits

Las imágenes de mapas de bits o rásters se pueden generar mediante dispositivos digitalizadores de imágenes como el escáner, la cámara digital o la misma computadora, al ejecutar la función de imprimir pantalla.



Saber más

La trayectoria que sigue la cámara digital para formar la imagen digital es la siguiente: la luz llega al sensor de imagen (denominado CCD, formado por multitudes de receptores fotosensibles), lo que provoca que se genere una señal eléctrica en cada receptor, esta señal se transforma en datos digitales por el convertor ADC, como una serie de cadenas de números ceros y unos (dígitos binarios) y por último, se representan los píxeles.



Estas imágenes digitales se crean formando una retícula rectangular compuesta por celdillas o cuadrículas. Cada una de las celdillas recibe el nombre de **píxel**. Este nombre proviene de la expresión que se obtiene al unir las palabras inglesas *picture* y *element*.

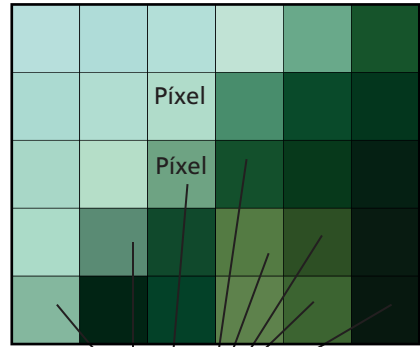


Definición

Los **píxeles** son las unidades de color que componen la imagen.

Un píxel solo puede ser de color rojo, azul o verde, o la mezcla de los tres, y puede tener en sí solo un color. Además, un píxel no tiene una medida concreta, no se puede decir si mide un milímetro o un centímetro o un metro, simplemente se puede afirmar que es la medida de división de una retícula en celdillas (figura 1.5).

Al visualizar todos los píxeles juntos, uno al lado del otro, dan la impresión de continuidad respecto a la tonalidad del color, lo que da lugar a la imagen:



Cada cuadradito es un píxel

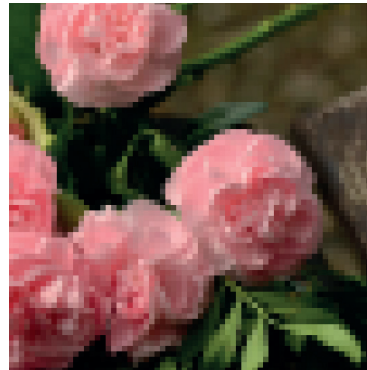
Fig. 1.5

Ejemplo

En la fotografía que se ilustra (figura 1.6 a) se puede observar que al aumentar su tamaño, se forma el cuadrado de mosaicos que componen la imagen (figura 1.6 b), donde los diferentes píxeles que la forman toman cada uno una tonalidad diferente de color.



a. Imagen original



b. Imagen ampliada

Fig. 1.6

Como se iba analizando, la imagen digital está formada por una matriz de píxeles ($a \times b \times c$) donde **a** y **b** representan la altura y anchura de la imagen y **c** la profundidad de color o profundidad de bit; se podría decir que es la tercera dimensión de la matriz, la que permite que cada píxel pueda tener un número determinado de colores distintos.

1.1.2.1 La profundidad de color



Definición

La **profundidad de color** se refiere al número de bits necesarios para representar y guardar la información de color de cada píxel en una imagen.

Cuanto mayor sea la profundidad de color en bits, la imagen dispondrá de una paleta de colores más amplia. Por tanto, la profundidad de píxel es una unidad de medida binaria, porque cada píxel está formado por bits; se puede decir entonces que una imagen “está a 24 bits”, cuando usa un total de 24 bits (16 millones de colores) de información por cada uno de los píxeles que la forma. Es decir: cuando se expresa que la profundidad de píxel es de 1 bit, la imagen solamente tiene dos colores o dos niveles de gris, representados por los valores 0 y 1. Una profundidad de píxel de 8 bits permite que cada píxel pueda tener 256 colores distintos o 256 niveles distintos de grises que provienen de combinar las series de ocho elementos posibles, obtenidas de combinar ceros y unos.

Luego, debes haber notado que el número de bits por píxel determinará la gama de colores de una imagen, según lo expresado en la tabla que se ofrece a continuación (tabla 1.1).

Tabla 1.1 Profundidad de color

No. de bit/canal	Número máximo de colores	
	Escala de grises	Color RGB
1 bit	2	8
2 bits	4	48
4 bits	16	4 096
8 bits	256	16,7 millones (24 bits)
12 bits	4 096	68 700 millones (36 bits)
14 bits	16 384	4,4 billones (42 bits)
16 bits	65 536	281 billones (48 bits)

Como se planteó anteriormente, cada píxel solo puede ser de un color, cuando se dice que una imagen es de 256 colores, se quiere decir que cada píxel puede tomar uno de esos 256 colores.

Si la profundidad es de 24 bits, se tiene la posibilidad de manejar millones de colores con la combinación de los tres primarios: rojo (**red**), verde (**green**) y azul (**blue**), es el llamado **modo RGB**. Cada canal de color tiene 8 bits para controlar cada color, por lo tanto, los colores que se pueden obtener son $256 \times 256 \times 256 = 16\,777\,216$; por eso, se llama “millones de colores” y se conoce como **color real**. Esta cantidad de colores es suficiente para que el ojo humano perciba las imágenes con calidad fotográfica realista.



Saber más

Se utiliza 1 bit para imágenes en blanco/negro, sin grises (0 = color **negro**, 1 = color **blanco**); 2 bits = 4 colores (00 = color **negro**, 01 = color **x**, 10 = color **y**, 11 = color **blanco**); 3 bits = 8 colores; 8 bits = 256 colores; 24 bits = 16,7 millones de colores.

Hasta ahora se ha visto cómo se almacena la información de color en los píxeles, pero no te has preguntado qué nos permite describir el color de cada uno de estos puntos o píxeles.

1.1.2.2 Los modos de color



Definición

Al sistema de coordenadas que nos permite describir el color de cada píxel, utilizando valores numéricos, se le llama **modo de color**.

Los modos de color más utilizados son el **RGB** (rojo, verde y azul); el **HLS** (tono, luminosidad, saturación) y el **CMYK** (cian, magenta, amarillo y negro).

Los modos de color afectan el tamaño de la imagen en disco y el número de canales de color que utilizan. Si se incrementa el número de colores, aumentará el tamaño del archivo que contiene la imagen. A continuación, se abordarán los diferentes modos de color y sus características:

- **Modo monocromático:** se corresponde con una profundidad de color de 1 bit. La imagen está formada por píxeles blancos o píxeles negros puros.

- Modo escala de grises: maneja el canal negro y permite 256 tonos de gris entre el blanco puro y negro puro, con una profundidad de 8 bits de color por píxel.
- Modo color indexado: utiliza un canal de color indexado de 8 bits, con el que se puede obtener hasta un máximo de 256 colores (2⁸). Es útil para aplicaciones multimedia.
- Modo RGB: es el acrónimo de **red** (rojo), **green** (verde), **blue** (azul). Cada color se forma por la combinación de tres canales, donde cada canal se corresponde con un color primario: rojo, verde o azul. En él se asigna un valor de intensidad de color que oscila entre 0 y 255. De la combinación de los tres canales con una profundidad de 24 bits por píxel, surgen hasta 16,7 millones de colores. Por ejemplo, el color rojo puro se representa por un valor de R: 255, G: 00 y B: 0. La suma de los tres colores da luz blanca.
- Modo CMYK: cada color se forma por la combinación de cuatro canales. Cada canal se corresponde con un color primario de impresión: **cyan** (ciano), **magent** (magenta), **yellow** (amarillo) y **black** (negro). Cada canal puede tener un valor entre 0 y 255. La profundidad de color de cada canal puede ser de 8 bits (lo normal), 16 o 32 bits.
- Modo HSB: en este modo, cada color surge de los valores de tres parámetros: **hue** (tono) que es el valor de color (rojo, verde, azul, etc.); **saturation** (saturación), que se refiere a la pureza del color y va del valor-100 al +100, **brightness** (brillo), que hace referencia a la intensidad de luz del color; es decir, la cantidad de negro o blanco que contienen estando su valor entre -100 (negro) y + 100 (blanco). La imagen en este modo se maneja con una paleta de colores de 24 bits, al igual que en el **modo RGB**.

Debes tener presente que cuando se trabaja con imágenes, estas pueden tener distintos destinos y de acuerdo con esto, será necesario seleccionar un modo de color:

- Si su destino es digital: pantalla, Web, apps, animación, videojuegos, etc., se utilizará el modo RGB.
- Si fuera impreso: soportes de papeles varios, distintos materiales de *packaging*, telas, etc., entonces se seleccionará el modo CMYK.

Es importante tomar esta decisión de acuerdo con la utilización de la imagen, ya que la forma en que se crean los colores es muy distinta: el formato digital se crea a través de rayos de luz y permite ver colores mucho más saturados, que son imposibles de conseguir de forma impresa (excepto empleando tintas especiales), ya que los colores se generan por medio de tintas que suprimen el paso de la luz. Además, los modos de color afectan el tamaño de la imagen en disco y el número de canales de color que utilizan, y si se incrementa el número de colores, aumentará el tamaño del archivo que contiene la imagen.

Te has preguntado alguna vez qué determina la calidad de una imagen. Pues bien, esto lo hace otro elemento importante en el trabajo con la imagen digital, que es la resolución.

1.1.2.3 El tamaño y la resolución



Definición

El **tamaño** de una imagen describe sus dimensiones físicas y viene definido por el producto de dos números enteros correspondientes al número de píxeles del ancho y del alto de la imagen.

El tamaño de la imagen puede expresarse de diferentes maneras:

- **Píxeles:** indicando la cantidad de píxeles horizontales y verticales que posee. Ejemplo: 640 x 480 (figura 1.7).
- **Centímetros:** indicando el tamaño horizontal y vertical que tendrá a la hora de imprimirla o visualizarla. Depende de la cantidad de píxeles y la resolución (que se abordará más adelante). Ejemplo: 20 cm x 15 cm (figura 1.8).
- **Bytes:** es la cantidad de memoria que ocupa según los datos que posee. Una imagen que ocupa 1,04 Mb, por ejemplo, tendrá más detalles que otra de 754 Kb y se podrá imprimir a mayor tamaño sin que se vea el píxel (figura 1.9).

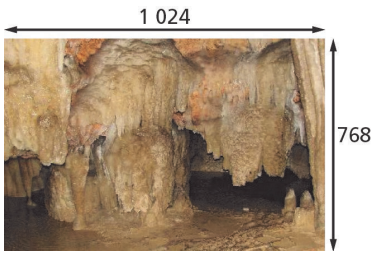


Fig. 1.7

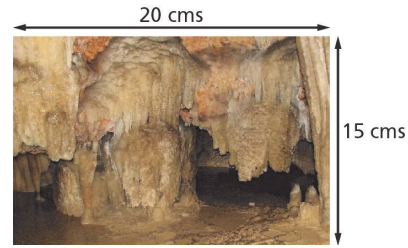


Fig. 1.8



Fig. 1.9

Por tanto, cuando alguien dice que tiene una imagen de 800 x 600 píxeles, uno puede hacerse una idea exacta de su tamaño. Pero en realidad se sabrá, ¿cuánto mide cada uno de los píxeles que componen esta imagen?

Como no puede ser de otro modo, la respuesta a esta interrogante depende de la resolución.



Definición

La **resolución** de una imagen es el grado de detalle o calidad de una imagen digital, ya sea escaneada, fotografiada o impresa. Se define como el número de píxeles que contiene por unidad de longitud (se utiliza normalmente la pulgada donde 1 pulgada = 2,54 cm).

Es decir, la resolución expresa la densidad de píxeles de la imagen. El valor de la resolución va acompañado, o sea, se expresa por el acrónimo **ppp** (píxeles por pulgada), que en ocasiones se encuentra en inglés **dpi** (*dots per inch*).

Como te habrás dado cuenta la resolución está estrechamente ligada al nivel de detalle de la fotografía: cuanto más píxeles contenga una imagen por pulgada lineal, mayor calidad tendrá, por tanto, mayor resolución y más posibilidades de aumentar el tamaño de la imagen sin perder calidad, pero no olvides que si aumentas la resolución de una imagen, esto afectará el espacio que ocupa al guardarla, ya que al aumentar el número de puntos por pulgada, se tendrá más información y la imagen ocupará más espacio en la memoria, por lo que pesará más.

A partir de una resolución de impresión de unos 200 ppp, se puede decir que la resolución es de buena calidad. Por ejemplo, si una imagen tiene una resolución de 200 ppp, esto significa que en cada pulgada cuadrada se encontrará la cantidad de 200 x 200 píxeles, lo que dará un resultado de 40 000 píxeles que contienen la información digital de esa imagen. Si esa misma imagen tiene una resolución de 72 ppp, la información digital está contenida solamente en 5 184 píxeles, con lo que la calidad de la imagen será menor, o sea, a menor resolución, menor información y menor detalle.

Para que puedas comprender mejor este aspecto, es necesario que conozcas la diferencia entre los siguientes conceptos: tamaño de la imagen, resolución de la imagen y tamaño del archivo.

El **tamaño de la imagen** depende de sus dimensiones (anchura y altura), medidas en píxeles u otras unidades: puntos, picas, centímetros, milímetros, pulgadas, etcétera.

La **resolución de la imagen** se expresa en píxeles por pulgada. A mayor resolución, más píxeles caben en una pulgada y menor será el tamaño de los píxeles.

El **tamaño del archivo** de la imagen es una cifra en bits o en bytes, que describe la cantidad de memoria necesaria para almacenar la información de la imagen en un soporte (disco duro, CD, tarjeta de memoria, etc.). Y, como ya te imaginas, el tamaño del archivo dependerá de varios factores, especialmente de la resolución (R), las dimensiones de la imagen (largo x ancho) y la profundidad de color (P). Puedes calcular el tamaño de un archivo con la siguiente fórmula:

$$\text{Tamaño} = (\text{resolución})^2 \times \text{largo} \times \text{ancho} \times \text{profundidad de color}$$

A continuación, se muestra un ejemplo de cómo se procede en el cálculo:

Una imagen de 10 cm x 15 cm (3,94 x 5,91 pulgadas), con una resolución de 96 ppp y una profundidad de color de 32 bits, tendrá un tamaño bruto de:

$$9\ 216 \cdot 3,94 \cdot 5,91 \cdot 32 = 68\ 657\ 144 \text{ bits}$$

es decir: 858 393 bytes

o, lo que es lo mismo: 838 kbytes

Por tanto, a mayor tamaño de la imagen (más píxeles contendrá), mayor será el tamaño del archivo.



Recuerda que...

Recuerda que 1 byte son 8 bits y que 1 kilobyte equivale a 1 024 bytes.

Ahora bien, seguro te preguntarás ¿cuándo utilizar una resolución u otra? Esto dependerá del destino final que tendrá la imagen sobre la que estés trabajando.

Si vas a utilizar las imágenes exclusivamente en la pantalla de la computadora, deberás tener en cuenta la resolución de la pantalla, y con 72 ppp o 72 dpi será suficiente, así podrás ver en su tamaño real, las imágenes en una pantalla con una resolución de 800 x 600 píxeles.

Debes saber que las imágenes de Internet suelen tener un máximo de 72 ppp, pues como los monitores no son capaces de mostrar una resolución mayor de impresión, no se hace necesario una resolución mayor a esta.



Definición

La **resolución de la pantalla o del monitor** se refiere al número de píxel por pulgada que es capaz de mostrar.

Es habitual medir la resolución de la pantalla por el total de píxeles que abarca, de ahí que se puedan encontrar diferentes resoluciones de pantalla atendiendo al número de columnas y filas de píxeles: 640 x 480; 800 x 600 y 1 024 x 768; las dos últimas son las más utilizadas.

Como en un mismo monitor se pueden encontrar diferentes resoluciones de pantalla, es importante saber que cuanto mayor sea la resolución de la pantalla, menor será el tamaño de los píxeles y mayor la definición de la imagen.



Saber más

Se suele decir que para ver una imagen en su tamaño real en una pantalla con resolución de 800 x 600, la resolución de la imagen debe ser 72 ppp; en una resolución de 1 024 x 768, debe tener 96 ppp.

Cuando se trabaja con una imagen que tiene una resolución mayor que la de la pantalla, esta se observa con un tamaño mayor al original, mientras que cuando se trabaja con una imagen con resolución menor que la pantalla, se observa con un tamaño menor del que tiene; como se muestra en la imagen que aparece a continuación.



300 ppp de resolución.

El tamaño de la imagen es de 1,25 x 1,25 pulgadas.



72 ppp de resolución.

El tamaño de la imagen es de 1,25 x 1,25 pulgadas.

Como puedes ver las imágenes tienen el mismo tamaño, pero se visualizan en forma diferente.

Para saber cuál es el tamaño máximo de impresión de una imagen de mapa de bits, se tiene que dividir el ancho y el alto (en píxeles) por la resolución (en píxeles por pulgada).

$$\text{Tamaño (pulgadas)} = \frac{\text{ancho o alto (píxeles)}}{\text{resolución (ppp)}}$$

$$\text{Resolución (ppp)} = \frac{\text{ancho (píxeles)}}{\text{longitud (pulgadas)}}$$

Ejemplo

1. Calcula el tamaño máximo de impresión (en centímetros) de una fotografía de resolución de impresión de 180 ppp y que tiene una calidad máxima de 3,648 x 2,736 píxeles.

Para obtener el tamaño de impresión, se divide el ancho y el alto por la resolución. Para convertir en centímetros, se debe tener en cuenta que 1 pulgada = 2,54 cm.

$$\text{horizontal} = \frac{\text{ancho}}{\text{resolución}} = \frac{3,648 \text{ píxeles}}{180 \text{ ppp}} = 20,26 \text{ pulgadas} \times 2,54 \frac{\text{cm}}{\text{pulgadas}} = 51,46 \text{ cm}$$

$$\text{vertical} = \frac{\text{ancho}}{\text{resolución}} = \frac{2,736 \text{ píxeles}}{180 \text{ ppp}} = 15,2 \text{ pulgadas} \times 2,54 \frac{\text{cm}}{\text{pulgadas}} = 38,6 \text{ cm}$$

2. Calcula la resolución total de la cámara digital con que se han obtenido las fotos (es decir, el número total de megapíxeles).

Para saber cuál es la resolución de la cámara digital, se debe conocer con cuántos píxeles de ancho y alto es capaz de obtener una imagen y multiplicarlos.

$$3,648 \times 2,736 = 9,980,928 \text{ píxeles} = 9,98 \text{ megapíxeles}$$

Repite las operaciones anteriores para una resolución de impresión de 200 ppp. ¿Por qué el tamaño de impresión es menor que el del ejemplo?

1.1.2.4 Formas de obtener o generar imágenes de mapa de bits

Cuando se habla de generar imágenes de bitmap o ráster, como también se conocen, se está haciendo referencia al dibujo en bitmap, donde se aplican colores a los puntos o píxeles que forman la imagen. Esta generación de imágenes bitmap se realiza con el empleo de diferentes herramientas, como pueden ser el GIMP o el Photoshop, en cualquiera de sus versiones. La obtención de estas imágenes digitales se puede hacer de varias formas, una de ellas es la Web (Internet), realizando búsquedas con Google, por ejemplo, o bien desde una Web especializada.



Recuerda que...

Las imágenes obtenidas de la red pueden estar sujetas a derecho de autor o copyright. Por tanto, se debe tener cuidado con su utilización y difusión pública.

La captura de pantalla es una de ellas, y consiste en crear una imagen a partir de lo que aparece en la pantalla de la computadora. Los procedimientos para realizar este tipo de captura son los siguientes:

- Para capturar una pantalla con GIMP, desde cualquier sistema operativo, hay que ir al menú **Archivo/Create/ Captura de pantalla**, y aparecerá un cuadro de diálogo en el que se puede elegir entre capturar toda la pantalla o una sola ventana, e incluso realizar cualquiera de estas dos operaciones tras un período de retardo que se puede configurar. Para capturar la pantalla, hay que pulsar el botón “Capturar”. Si se escoge la opción de capturar una sola ventana, se tendrá que indicar **cuál** por medio del puntero del ratón. La imagen de mapa de bits generada se puede guardar en varios formatos diferentes. Si se pretende trabajar después sobre esta imagen con GIMP (por ejemplo, aplicando herramientas como diferentes capas), se tendrá que guardar en formato XCF, que es el formato de imagen propio del programa.
- En Windows hay también otro procedimiento para capturar pantallas que no requiere tener instalado ningún programa adicional. Lo único que se necesita es el programa **Paint** que, al ser un accesorio de Windows, viene instalado con el propio sistema operativo. Para capturar una pantalla con Windows no hay más que pulsar la tecla “Impr Pant I PetSis” o “PrtSc Sys Rq” que está en todos los teclados. Al pulsarla, se le está indicando

al sistema operativo que transforme la información de la pantalla en una imagen digital y que la conserve en el **Portapapeles** como si de una fotografía se tratase. El **Portapapeles** es una parte de la memoria RAM (Memoria de Acceso Aleatorio) que el sistema operativo utiliza para pasar datos de un programa a otro. Una vez capturada la pantalla, abriremos **Paint**, si no está abierto ya, y se escoge en el menú **Edición/Pegar**, con lo que aparecerá en el área de trabajo de **Paint** el “pantallazo” que hemos generado, y que ahora se podrá tratar como una foto digital en mapa de bits.

- Si se está trabajando sobre **Ubuntu**, este sistema operativo cuenta con una herramienta específica para capturar pantallas que suelen tener un icono de acceso rápido en la barra de tareas del escritorio.

El **escanear imágenes** es otra forma de obtener imágenes digitales. Un escáner es un periférico que se utiliza para transformar imágenes impresas en imágenes digitales. Su nombre viene del término en inglés *scanner*, que proviene a su vez del verbo **to scan**, que significa “escrutar, echar un vistazo, recorrer con la vista”. El escáner tiene una lámpara que “lee” lo que se le pone delante, al igual que ocurre con las fotocopadoras, pero en lugar de reproducir la imagen de nuevo en papel, transforma la información de la imagen en dígitos binarios, por eso se dice que **digitaliza** la imagen impresa.

Los escáneres más habituales hoy en día son los de sobremesa, pero los hay de mano y de rodillo. Desde hace varios años se están extendiendo mucho las impresoras multifunción (que incorporan además fax y escáner).

Por último, no se deben olvidar las **cámaras digitales**, que también permiten obtener dicho tipo de imagen al descargar en la PC las imágenes digitales obtenidas.

1.1.3 Los formatos de la imagen digital

Una vez creada nuestra imagen, ya sea capturada con la cámara o creada a mano, se debe guardar en un archivo. El archivo, con un nombre y una extensión, no solo contiene la información de cada píxel, sino también una cabecera en la que se guarda información destinada al programa, encargada de abrir la imagen y mostrarla en el monitor. Este archivo que se crea tiene un formato gráfico.



Definición

Un **formato gráfico** define la manera en que se guarda una imagen y toda la información relacionada con esta.

Las imágenes digitales se pueden guardar en distintos formatos. Cada formato tiene sus ventajas y desventajas. Algunos admiten solo hasta 256 colores, otros una gran compresión. Cada uno se corresponde con una extensión específica del archivo que lo contiene. A continuación, se verán los más frecuentes (tabla 1.2), aunque hay otros, como: .raw, .psp, .pcx, .tga, etcétera.

Tabla 1.2 Principales formatos de mapas de bits

Formato	Descripción	Extensión
BMP <i>(Bitmap = Mapa de bits)</i>	- Formato de calidad, en el que la imagen se forma mediante una parrilla de píxeles. Este formato no sufre pérdidas de calidad y, por tanto, resulta adecuado para guardar imágenes que se desean manipular posteriormente. - Tiene como ventaja que guarda gran cantidad de información de la imagen, pero tiene como inconveniente que el archivo tiene un tamaño muy grande.	.bmp
TIFF <i>(Tagged Image File Format = Formato de Archivo de Imagen Etiquetada)</i>	- Almacena imágenes de una calidad excelente. Utiliza cualquier profundidad de color de 1 bit a 32 bits. Es el formato ideal para editar imágenes de alta calidad que van a ser impresas. - Tiene como ventaja que es ideal para conservar archivos originales, pero como inconveniente, que produce archivos muy grandes.	.tif

GIF <i>(Graphics Interchange Format = Formato de Intercambio Gráfico)</i>	■ Ha sido diseñado específicamente para comprimir imágenes digitales. Reduce la paleta de colores a 256 colores como máximo (profundidad de color de 8 bits). Admite gamas de menor número de colores, y esto permite optimizar el tamaño del archivo que contiene la imagen. ■ Tiene como ventaja que es un formato idóneo para publicar imágenes en internet, pudiendo comprimir las imágenes sin pérdidas. Utiliza el modo de color indexado para las imágenes que no tienen muchas tonalidades de color. Permite gráficos animados y transparencia. ■ Pero el inconveniente está dado en que no es recomendable para fotografías de cierta calidad ni originales, ya que el color real o verdadero utiliza una paleta de más de 256 colores.	.gif
JPG <i>(Joint Photographic Experts Group = Grupo de Expertos Fotográficos Unidos)</i>	■ Es el formato más utilizado en internet para la reproducción de fotografías. Permite comprimir las imágenes, pero produce pérdidas de calidad. En la mayoría de los casos esta pérdida se puede asumir porque permite reducir el tamaño del archivo y su visualización es aceptable. Es recomendable utilizar una calidad del 60 % - 90 % del original. ■ A diferencia del formato GIF, admite una paleta de hasta 16 millones de colores. ■ Las cámaras digitales suelen almacenar directamente las imágenes en formato JPEG, con alta calidad. ■ Tiene como ventaja que es ideal para publicar fotografías en la Web, siempre y cuando se configuren adecuadamente	.jpg

Formato	Descripción	Extensión
	dimensiones y compresión. Pero tiene como inconveniente que al definirse un factor de compresión, se pierde calidad, por lo que no es recomendable para archivar originales.	
PNG (<i>Portable Network Graphic = Gráfico portable para la red</i>)	Es un formato alternativo al GIF que tiene las ventajas de los formatos GIF y JPG. Comienza a ser muy utilizado en Internet por su gran capacidad de compresión, sin pérdida y con posibilidades de transparencia. Tiene una tasa de compresión superior al formato GIF (+ 10 %), y admite la posibilidad de emplear un número de colores superior a los 256 que impone el GIF.	.png
PICT (<i>SoftImage PICT file</i>)	Es el formato característico de la plataforma MAC. Permite ser comprimido sin perder calidad de imagen.	.pic

Es válido aclarar que el **.psd** es el archivo fuente de Adobe Photoshop, cuyo fin es, luego de terminar el diseño, convertirlo a algunos de los formatos mencionados; de igual manera sucede con la extensión **.xcf**, formato nativo de GIMP (*software* libre).



Recuerda que...

Para visualizar la **extensión de los archivos**, debes tener deshabilitada la opción de "Ocultar las extensiones de archivo para tipos de archivo conocidos" en tu sistema operativo.

Para reducir el tamaño de los archivos de imagen y hacerlos más manejables, casi todas las cámaras digitales usan alguna forma de compresión. Para ti es importante conocer el formato del archivo con el que la cámara guardará las fotografías realizadas, ya que de él y de su sistema de compresión, dependerá la calidad de estas una vez que las guardes en la computadora o las imprimas.



Saber más

A la hora de guardar las imágenes, los distintos formatos utilizan diferentes tipos de compresión. Hay dos modos de compresión de archivos de imagen:

Compresión sin pérdidas: se utilizan algoritmos de compresión, pero no se pierde ningún dato, ya que se comprime muy poco la imagen para que su calidad sea semejante a la fuente original. Ejemplos son los formatos RAW, TIFF, GIF o BMP.

Compresión con pérdidas: en los formatos de compresión con pérdidas se aplican algoritmos que permiten decidir cuál es la información menos relevante para el ojo humano y la desechan suprimiendo datos o guardando valores de píxeles semejantes como uno solo. Esta compresión degrada la imagen en proporción al grado de compresión. Cuanta mayor cantidad de datos se desechen, menos ocupa el archivo, pero también tiene una calidad menor. Al descomprimirla, puede parecer visualmente “normal” si no se amplía en exceso. Los formatos más usados son el JPEG y el PNG.

Como recordarás, las imágenes vectoriales se representan con trazos geométricos que están controlados por operaciones matemáticas que realiza la computadora. Las líneas que componen la imagen están definidas por vectores (de ahí su nombre). La ventaja de este tipo de imágenes es que pueden reducirse o ampliarse sin ningún tipo de pérdida de calidad. Los principales formatos de las imágenes vectoriales se muestran en la tabla 1.3.

Tabla 1.3 Principales formatos de las imágenes vectoriales

Formato	Descripción	Extensión
<i>SVG</i>	Vectores gráficos escalables. Formato muy reciente, pero que puede ser usado fácilmente en GIMP.	.svg
<i>WMF</i>	Es el formato que mejor se adapta a los distintos programas de dibujo.	.wmf
<i>EPS</i>	Formato utilizado habitualmente para exportar imágenes de mapa de bits con trazados vectoriales.	.eps

1.1.4 La composición, el retoque y la optimización de la imagen digital

Una vez que se tenga una imagen, ya sea capturada con la cámara o creada a mano, se pueden realizar diferentes acciones con ella, como editarla, retocarla o componerla.

La composición y la edición de imágenes es, sobre todo, una actividad divertida y gratificante, que potencia las capacidades de creación y comunicación de las personas. Sin embargo, el **retoque fotográfico** es una técnica que te permitirá obtener una nueva imagen modificada, ya sea para lograr una mejor calidad, más efectismo, o para obtener una composición totalmente diferente que distorsione la realidad, con el objetivo de conseguir efectos impactantes o simplemente corregir diversos errores en las imágenes originales. Para llevar a cabo dicho proceso, se utilizan programas informáticos.

Las técnicas de retoque fotográfico digital son hoy en día muy empleadas como método de postproducción, sobre todo en ámbitos donde la imagen es lo que vende. Se ve mucho su empleo en las producciones de modelaje, ya que la perfección en los modelos debe ser alcanzada a toda costa.

También en los procesos de publicidad cumplen una parte muy importante en el trabajo final, pues muchas veces uno se encuentra con fotografías que no son obtenibles solamente utilizando una cámara, y para esto se recurre al **fotomontaje o retoque digital**, en donde se pueden manipular varias fotografías para crear una composición final. En ocasiones, esto suele llamarse **hiperrealidad**.

Los fotomontajes sirven para obtener imágenes que por algún motivo no se pueden capturar naturalmente con una cámara, es decir, que hay que hacerlo de esta manera para darle efecto a las imágenes, porque no se captura lo que uno desea. Se puede decir que es una composición que se realiza mediante recortes de otras ilustraciones, y al final queda simplemente como una fotografía normal.

Bueno, hasta aquí se ha referido todo lo relacionado con la imagen digital fija, sus características y las diferentes formas para su obtención, por

lo que ya estás más preparado para ampliar con investigaciones tus conocimientos y profundizar en los contenidos que te resultaron más interesantes.

1.1.5 Los sistemas para el trabajo de la imagen digital

Las imágenes son editadas para optimizarlas, manipularlas, retocarlas, etc., con el fin de alcanzar alguna meta deseada, como puede ser: eliminar fallas que se pueden haber producido durante el escaneo o al tomar las fotografías por sobreexposición, por un contraste inadecuado, presencia de ruido, baja exposición o el efecto de ojos rojos.



Definición

Se le llama **editor de imágenes** a la aplicación con la cual se realiza la edición digital de imágenes. Este ofrece numerosas funciones en un menú o en una ventana de herramientas.

Algunas de las funciones que tiene el **editor de imágenes** son: ajustar el tiempo de exposición (dígase aclarar u oscurecer la imagen), realizar selecciones donde esta selección puede ser tratada independientemente del resto, ofrecer procedimientos automáticos de corrección para aquellas personas que no son muy conocedores de la materia; realizar cambios a los modos de color, la profundidad, el tamaño de la imagen, al formato, entre otros; también favorece la utilización de filtros y la aplicación de efectos a las imágenes, la realización de fotomontajes, el retoque fotográfico y la incorporación de textos.

La edición digital de imágenes es cada día más utilizada, ya que la fotografía digital permite transferir las fotos directamente a la computadora y realizar un tratamiento básico a partir de las opciones automáticas, o un tratamiento especializado como el que realizan los fotógrafos y diseñadores en la autoedición y en la preimpresión.

Las imágenes modificadas, por cualquiera de los editores de imágenes digitales, se pueden emplear posteriormente en publicaciones como: revistas, catálogos, libros y otros medios impresos y electrónicos donde aparecerán los rostros sin imperfecciones, la piel inmaculada, paisajes preciosos y otras características, que son consecuencias de la elaboración digital. También es importante conocer que la edición de imágenes es una

posibilidad de desarrollo artístico y así es considerada actualmente en el arte digital y en la multimedia.



Observación

No debes olvidar que la edición de imágenes digitales se puede utilizar intencionalmente para la manipulación de imágenes, por lo que siempre se debe emplear con fines bien definidos, donde se expresen ideas, con ética y profesionalismo.

Como se abordó anteriormente, existen diferentes tipos de imágenes digitales y para trabajar con cada una de ellas se hace necesario contar con diferentes sistemas, pero la selección de la aplicación que se va a utilizar dependerá del tipo de imagen con la que vayas a trabajar.

Cuando se necesite trabajar con imágenes vectoriales, se tendrán varios programas a nuestra disposición, tanto propietarios como de código abierto, entre ellos están los que se recogen en la tabla 1.4.

Tabla 1.4

Software propietarios	Software de código abierto
Illustrator, de Adobe	OpenOffice.org Draw, de Oracle
Freehand, de Macromedia	Xara LX
Corel Draw!, de Corel Corporation	LibreOffice Draw, del consorcio The Document Foundation
Visio, de Microsoft Corporation	Inkscape

Si se hace referencia al trabajo con imágenes de mapa de bits (bitmap), también se dispondrá de varios **software**, tanto propietarios como de código abierto. Algunos de ellos se recogen en la tabla 1.5.

Tabla 1.5

Software	Descripción	Tipo de software
<i>GIMP</i>	Muy potente, en continuo desarrollo.	Libre
<i>PhotoScape</i>	También usado como visor de imágenes.	Gratuito
<i>Photoshop</i>	Muy potente. Es el más usado	Propietario
<i>Paint Shop Pro</i>	De la familia de Corel	Propietario
<i>Pixel Image Editor</i>	No muy potente	Propietario
<i>Polar</i>	No muy potente	Libre

Para la visualización de las imágenes, contamos con los siguientes **software** de la tabla 1.6.

Tabla 1.6

Software	Descripción	Tipo de software
<i>XnView</i>	Programa para la visualización de fotos. También usado como organizador de fotos.	Libre
<i>Picasa</i>	Visualizador y editor de imágenes. Permite almacenar y compartir en la red.	Gratuito

Además de los programas de edición gráfica, existen Webs que nos permiten editar imágenes *online* con funciones muy parecidas. Estas web, cada vez más, se pueden encontrar con un gran número de opciones avanzadas, lo que nos ahorra tener que instalar cualquier programa. Muchas

de ellas están disponibles para dispositivos móviles: puedes descargar las apps desde la App Store (Apple) o Google Play (Android). Ejemplo de estas Web son las que aparecen en la tabla 1.7.

Tabla 1.7

Sitio Web	Características
<i>Pixlr</i>	Muy completo. Tiene tres variantes: Editor (muy completo y profesional, con gran cantidad de opciones); Express (más básico y para retoques rápidos); y O-Matic (que aplica retoques, marcos, filtros y efectos a toda la imagen de forma automática). Esta web no requiere registrarse.
<i>Phixr</i>	De interfaz sencilla, permite aplicar múltiples efectos. No requiere registrarse.
<i>PicMonkey</i>	Muy intuitivo (no se requieren muchos conocimientos para manejarlo), permite mejorar imágenes, crear <i>collages</i> , tarjetas, marcos, etc. No requiere registrarse.
<i>Photoshop Express Editor</i>	Parecido al programa original Photoshop, permite trabajar con capas. Solo admite imágenes en formato JPG, JPEG o JPE de hasta 16 Mb. No requiere registrarse.
<i>BeFunky</i>	Sirve para crear y editar <i>collages</i> . Es muy sencillo de usar e incluye múltiples filtros, texturas y efectos. No requiere registrarse.
<i>PicMagic</i>	Editor básico que permite aplicar efectos y filtros. No requiere registrarse.
<i>Fotor</i>	Permite hacer portadas, <i>collages</i> , etc. No requiere registrarse.
<i>FotoFlexer</i>	Muy intuitivo y fácil de usar, permite múltiples opciones de edición, filtros, etc. No requiere registrarse.
<i>Pho.to</i>	Con opciones limitadas, permite mejorar fotos. No requiere registrarse.

A continuación se estudiarán dos aplicaciones para el tratamiento de las imágenes digitales, una representante del *software propietario*, que será **Photoshop de Adobe**, y otra como representante del *software Open Source* (código abierto) que será **GIMP**.

1.1.6 La edición, el fotomontaje y el retoque de imágenes digitales

Comencemos analizando algunas de las características de los dos editores de imágenes que se abordarán en este libro y que son los más utilizados para la optimización, el retoque y la composición de imágenes digitales en el mundo actual.

Adobe Photoshop



Adobe Photoshop (popularmente conocido solo por su segundo nombre, Photoshop) es el nombre de uno de los programas más famosos de la casa Adobe. Se trata esencialmente de una aplicación informática en forma de taller de pintura y fotografía que trabaja sobre un “lienzo” y que está destinado a la edición, retoque fotográfico y pintura a base de imágenes de mapa de bits. Su nombre en español significa literalmente “taller de fotos”.

Photoshop soporta muchos tipos de archivos de imágenes, como son:

- PSD, PDD: formato estándar de Photoshop con soporte de capas.
- PostScript: no es exactamente un formato, sino un lenguaje de descripción de páginas. Se suelen encontrar documentos en PostScript. Utiliza primitivas de dibujo en su edición.
- EPS: es una versión de PostScript, se utiliza para situar imágenes en un documento. Es compatible con programas vectoriales y de autoedición.
- BMP: formato estándar de Windows.
- GIF: muy utilizado para las Web. Permite almacenar un canal alfa para dotarlo de transparencia y salvarlo como entrelazado, para que al cargarlo en la Web lo haga en varios pasos. Admite hasta 256 colores.
- JPEG: también muy utilizado en la Web, tiene un factor de compresión muy alto y buena calidad de imagen.
- TIFF: una solución creada para pasar de PC a MAC y viceversa.
- PICT: desde plataformas MAC se exporta a programas de autoedición como QuarkXPress.

- PNG: la misma utilización que los GIF, pero con mayor calidad. Soporta transparencia y colores a 24 bits. Solo las versiones recientes de navegadores pueden soportarlo.
- PDF: formato original de Acrobat. Permite almacenar imágenes vectoriales y de mapa de bits.
- RAW: formato estándar para cualquier plataforma o programa gráfico.

Esta aplicación permite el tratado de imágenes en capas, para poder modificar cada objeto de la imagen de forma totalmente independiente. La imagen final puede guardarse en el formato original .psd de Photoshop, que soporta capas, o en un formato plano sin capas que puede ser .png, .bmp, .jpg, .gif, .pdf, etcétera.

Photoshop al igual que la mayoría de las aplicaciones para el tratamiento de la imagen fija cuenta con muchas herramientas que se emplean para el retoque y edición de imágenes y otras tareas más especializadas, estas son las siguientes:

- Herramientas de selección (rectangular, esférica, lazo manual, varita mágica, entre otras).
- Herramientas de pintado como: pincel, brocha, aerógrafo, relleno, texturas, degradados, etcétera.
- Conjunto extensible de pinceles para ser utilizados con las herramientas anteriores.
- Herramientas de modificación de escala, de inclinación, de deformación, de clonado plano, de clonado en perspectiva y otra de curado (para corregir pequeños defectos).
- Herramienta de creación y manipulación de texto.
- Creación y edición de degradados de colores.
- Herramientas o filtros para la manipulación de los colores y el aspecto de las imágenes, como enfoque y desenfoco, sombras, etcétera.
- Creación, edición y manipulación de máscaras específicas de una capa.
- Menú para deshacer todas las manipulaciones realizadas en una sesión (salvo las que sean irreversibles), para corregir errores o hacer pruebas.

De manera general, estas son las principales características de Photoshop. Ahora se pasará a conocer las de GIMP.

GIMP



Se identifica con la carita de un zorro llamado Wilber que es la mascota oficial del proyecto GIMP. Fue creado el 25 de septiembre de 1997 por Toumas, más conocido como **Tigert**.

GIMP (**GNU Image Manipulation Program**) es un programa de edición de imágenes digitales en forma de mapa de bits, tanto de dibujos como de fotografías. Un programa libre y gratuito que forma parte del proyecto GNU y está disponible bajo la licencia pública general de GNU. Puedes descargar gratuitamente dicho programa desde www.gimp.org.

Es uno de los programas empleados para la manipulación de gráficos que se puede utilizar en más sistemas operativos (Unix, GNU/Linux, FreeBSD, Solaris, Microsoft Windows y Mac OS X, entre otros). Su interfaz está disponible en varios idiomas, entre los que se encuentran: español, inglés (el idioma original), catalán, gallego, euskera, alemán, francés, italiano, ruso, sueco, noruego, coreano y neerlandés.

Esta aplicación tiene herramientas que se emplean para el retoque y edición de imágenes, es decir, para cambiar el tamaño, recortar, realizar el dibujo de formas libres, hacer fotomontajes, convertir a diferentes formatos de imagen, y otras tareas más especializadas. Se pueden crear imágenes animadas en formato GIF e imágenes animadas en formato MPEG, usando un **plugin** de animación.

También permite el tratado de imágenes en capas para poder modificar cada objeto de la imagen de forma totalmente independiente. La imagen final puede guardarse en el formato original .xcf de GIMP, que soporta capas, o en un formato plano sin capas que puede ser .png, .bmp, .jpg, .gif, .pdf, etcétera.

GIMP, como la mayoría de los programas para el tratamiento de imágenes estáticas, cuenta con muchas **herramientas**, entre las que se encuentran las siguientes:

- Herramientas de selección (rectangular, esférica, lazo manual, varita mágica, por color, inversión, yuxtaposición, adición, eliminación).
- Tijeras inteligentes.
- Herramientas de pintado como: pincel, brocha, aerógrafo, relleno, texturas, degradados, etcétera.
- Conjunto extensible de pinceles para ser utilizados con las herramientas anteriores.

- Herramientas de modificación de escala, de inclinación, de deformación, de clonado plano, de clonado en perspectiva y brocha de curado (para corregir pequeños defectos).
- Herramienta de creación y manipulación de texto.
- Creación y edición de degradados de colores.
- Posee también muchas herramientas o filtros para la manipulación de los colores y el aspecto de las imágenes, como enfoque y desenfoque, eliminación o adición de manchas, sombras, mapeado de colores, etcétera.
- También posee un menú con un catálogo de efectos y tratamientos de las imágenes.
- Asistentes para la creación rápida de botones, logos y otras imágenes simples, a partir de elementos preestablecidos.
- Creación, edición y manipulación de máscaras específicas de una capa.
- Creación, edición y manipulación de rutas.
- Medición y cálculo simple de longitudes y ángulos.
- Menú para deshacer todas las manipulaciones realizadas en una sesión (salvo las que sean irreversibles), para corregir errores o hacer pruebas.

Veamos ahora la forma de acceder a estas aplicaciones:

Photoshop	GIMP
1. Botón Inicio/Todos los programas/ Adobe Photoshop. 2. A partir del ícono de acceso directo que se ubica generalmente en el escritorio durante la instalación del programa.	1. Botón Inicio/Todos los programas/ GIMP. 2. A partir del ícono de acceso directo que se ubica generalmente en el escritorio durante la instalación del programa.

En ambos casos el usuario tendrá la posibilidad de seleccionar la forma de ver el entorno de trabajo que puede ser: como ventana única o paneles flotantes.

1.1.6.1 Entorno de trabajo de Photoshop y GIMP

Photoshop

El entorno de trabajo de Photoshop (el cual puede variar de acuerdo con la versión en cuanto a su aspecto, pero no en cuanto a sus principales componentes) presenta los siguientes elementos (figura 1.10):

1. Ventana de documento en forma de ficha.
2. Barra de menús.
3. Conmutador de espacios de trabajo.
4. Barra de título de panel.
5. Panel de control.
6. Panel de herramientas.
7. Botón contraer íconos.
8. Paneles acoplados verticalmente (panel de capas, canales y trazado; panel de ajustes y estilos; panel de color y muestras).

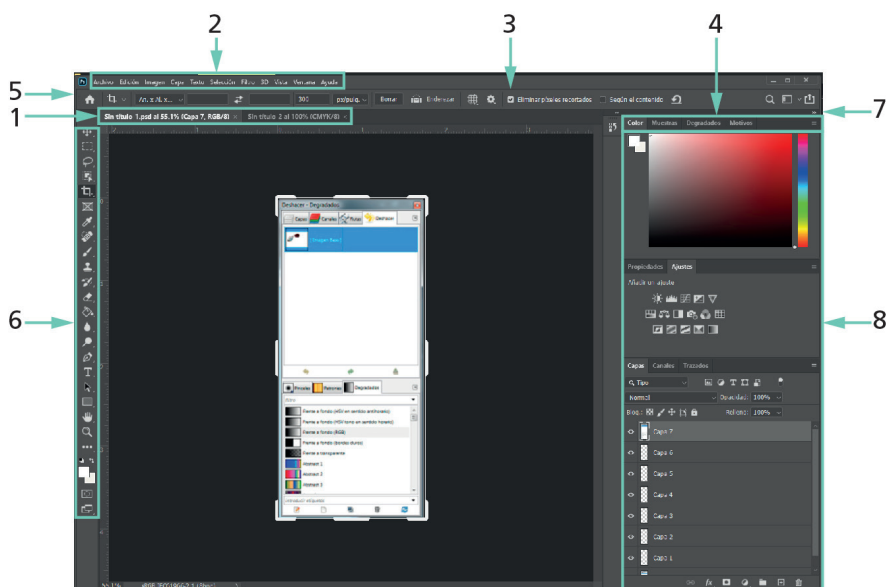


Fig 1.10

GIMP

En cualquiera de las modalidades en las que se encuentre el entorno de trabajo de la aplicación, se van a encontrar tres ventanas, que son las siguientes (figura 1.11):

1. Ventana de herramientas: esta contiene todas las herramientas de selección, pintura, de transformación, de color, etc., y las opciones de las herramientas seleccionadas.
2. Ventana de la imagen o de edición: es la ventana principal o de trabajo, posee un menú principal donde se pueden encontrar las siguientes pestañas: archivo, editar, seleccionar, vistas, imagen, capa, colores, herramientas, filtros, ventanas y ayuda.

3. Ventana de paneles acoplados verticalmente (panel de capas, canales, rutas, deshacer; panel de pinceles, patrones y degradados).

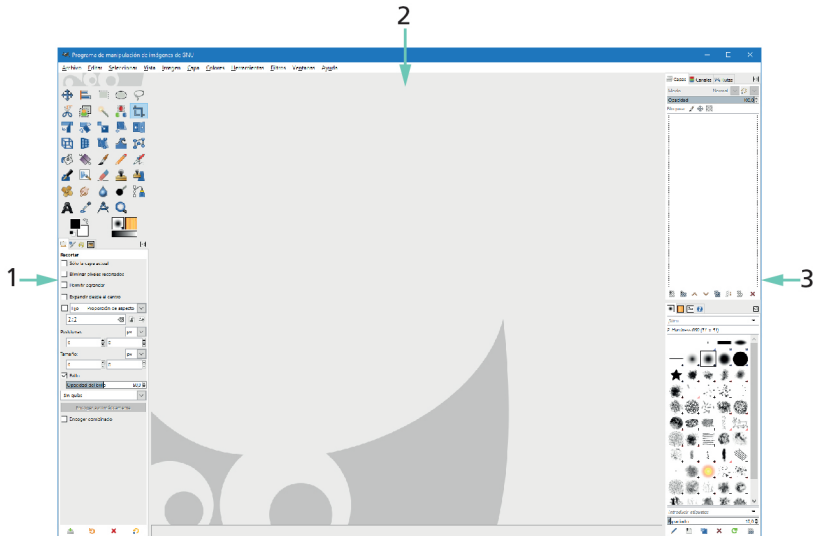


Fig. 1.11

Observación

No cierres de forma independiente la ventana de capas, canales y rutas, porque desaparecerá en futuras puestas en marcha del programa.

Las barras de herramientas

Como siempre, se comenzará analizando la barra de herramientas de Photoshop, dentro de la cual se encuentran las opciones de la figura 1.12.

Después se propone continuar con el estudio de la ventana de herramientas de GIMP, la que se compone de dos partes. La primera abarca la caja de herramientas propiamente dicha, que pone a disposición del operador las principales herramientas del programa, donde la que está siendo utilizada aparecerá siempre rodeada de un rectángulo y el acceso a la paleta de colores mostrará los colores seleccionados para el frente y el fondo (figura 1.13). La segunda, son las opciones de la herramienta seleccionada; en ella se muestran las variables que se pueden determinar para trabajar con esta herramienta. Esta ventana cambia cada vez que se elige una herramienta (figura. 1.14).

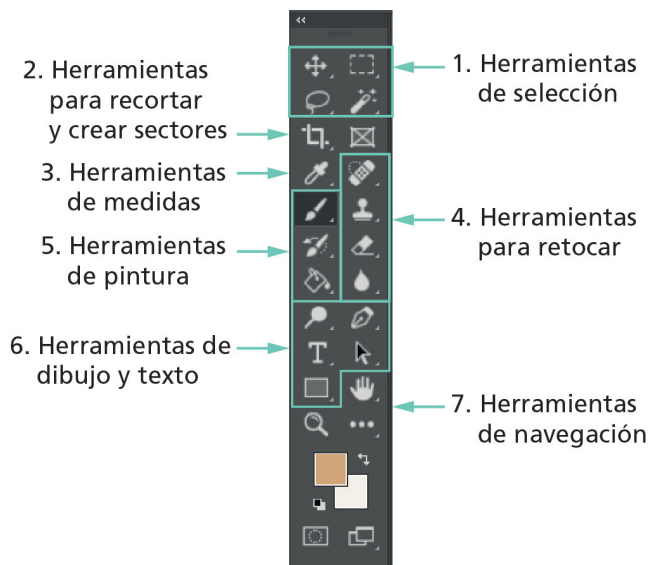


Fig. 1.12

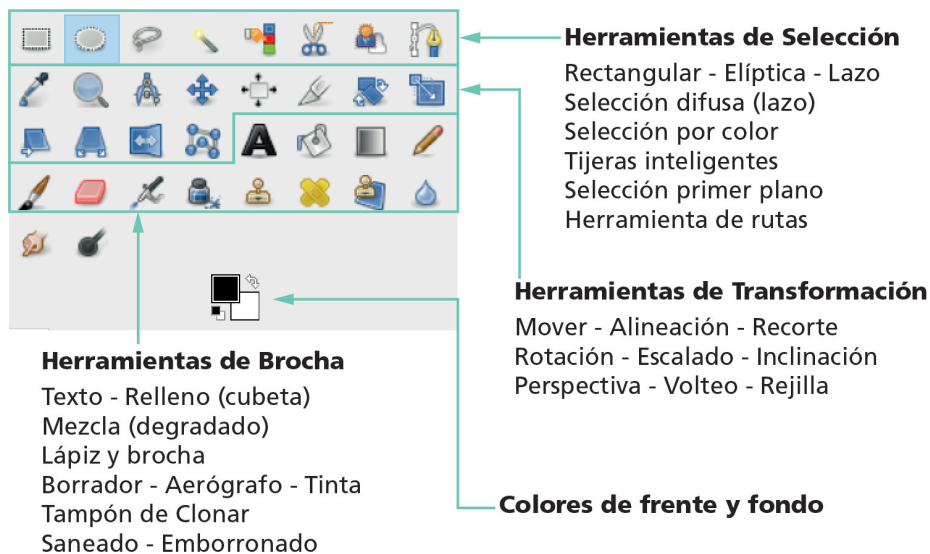
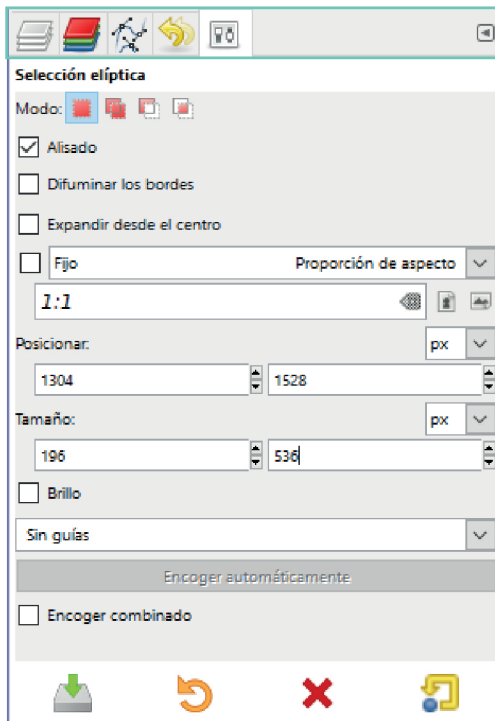


Fig. 1.13



Pestañas:

Capas, canales, deshacer, rutas, herramientas, etcétera.

Opciones de herramientas

Fig. 1.14

A B C **Cómo se hace**

- Si se quieren obtener los valores iniciales de la herramienta seleccionada, se debe hacer clic en el botón. 
- Para guardar los valores de las opciones, se presiona.
- Para recuperar o borrar los valores, se pulsa.

Como te habrás dado cuenta al analizar las herramientas de estas aplicaciones, se puede generalizar que en la mayoría de los programas para el tratamiento de las imágenes digitales, se van a encontrar herramientas para:

- Realizar diferentes tipos de selecciones.
- Trabajar con texto y hacer dibujos.
- Aplicar relleno, tanto de degradados como de un solo color.
- Realizar recortes y medir.
- Pintar o retocar.

Los menús de ambas aplicaciones

Se pasará ahora a estudiar los menús, pero esta vez lo haremos en paralelo para que te des cuenta de todas las opciones que tienen en común estas aplicaciones y la mayoría de los programas que se utilizan para la edición de imágenes digitales.

Se comenzará por el menú **Archivo (File)**, que es el primero que aparece en ambas aplicaciones (figuras 1.15 y 1.16).

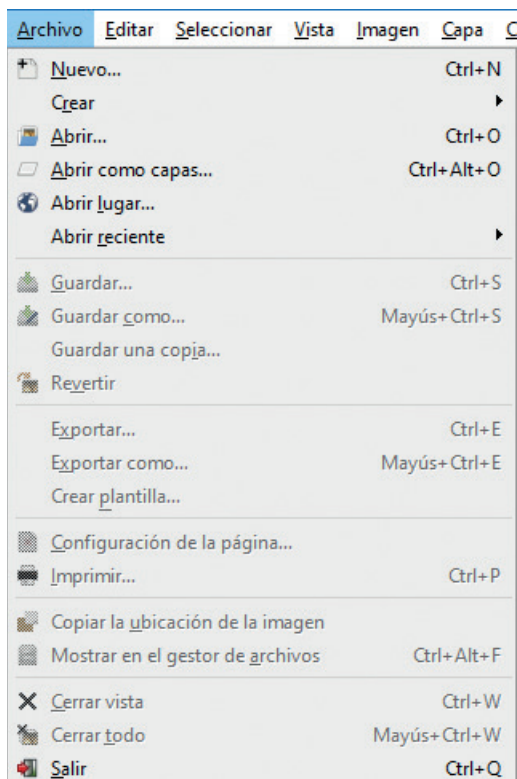


Fig. 1.15 GIMP.

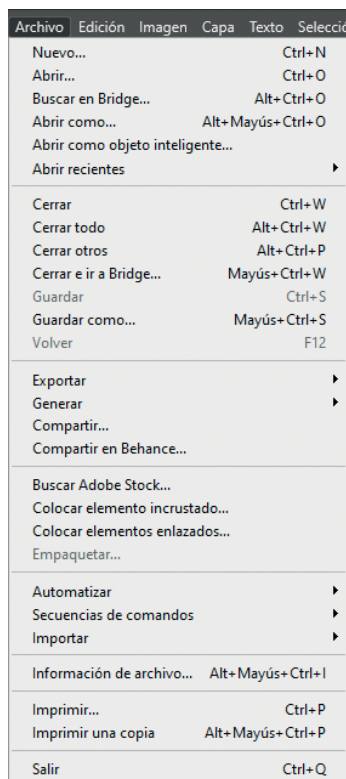


Fig. 1.16 Photoshop.

En el caso de GIMP, este menú permite crear archivos nuevos, abrir los existentes y los trabajados recientemente; guardar, imprimir y cerrar. En el caso de Photoshop, también cuenta con opciones para crear archivos nuevos, abrir los existentes, así como los recientes; guardar, imprimir y cerrar; pero además permite importar y exportar imágenes, automatizar algunas acciones y realizar ajustes a las páginas, en cuanto a la configuración.

Como segundo menú, aparece en ambas aplicaciones el menú **Edición** (**Edit**) (figuras 1.17 y 1.18). Como verás, en ambos menús aparecen las opciones para rehacer y deshacer, para cortar, copiar y pegar, rellenar, contornear y ajustar preferencias. Los cambios están dados sobre todo en Photoshop, donde se encuentra en este menú la posibilidad de realizar transformaciones libres u otras transformaciones en la imagen, definir pinceles, formas y motivos.

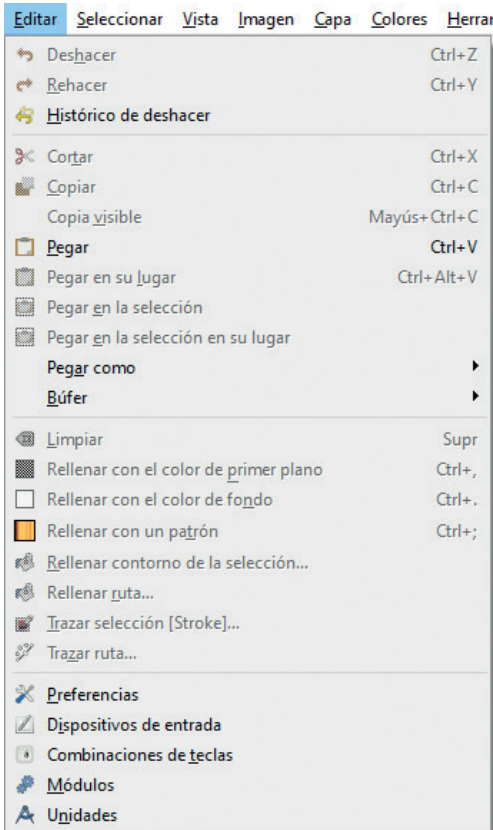


Fig. 1.17 GIMP.

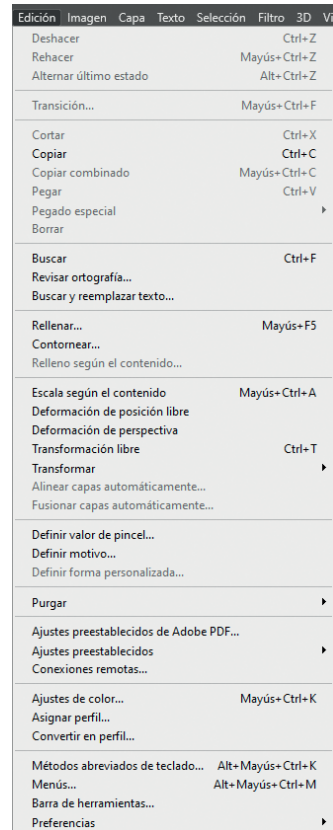


Fig. 1.18 Photoshop.

Ahora se verá el menú **Selección**, que también existe en ambas aplicaciones (figuras 1.19 y 1.20). En él podrás notar que comienzan con la opción "Todo", además están las opciones de modificar, seleccionar por

color, deseleccionar, invertir, guardar selección o cargar selección. En el caso de Photoshop, también tiene la opción calar, extender y cargar selección, mientras que GIMP permite guardar por canales.

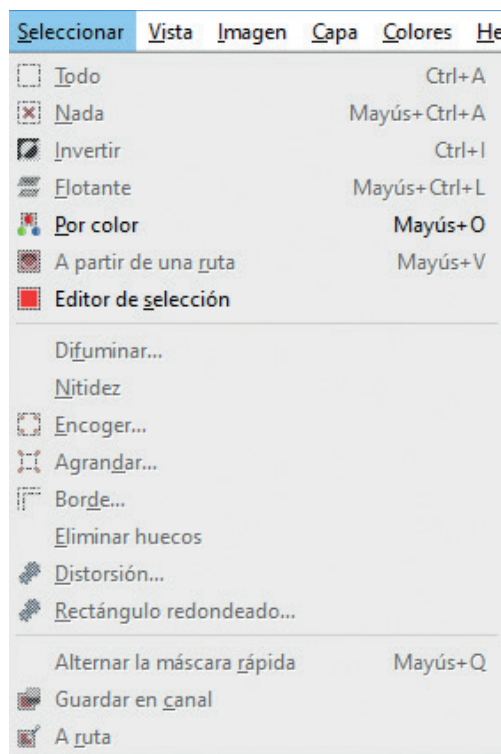


Fig. 1.19 GIMP.

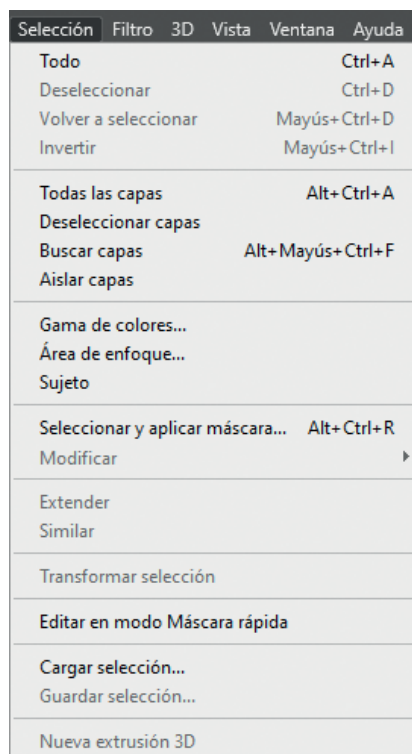


Fig. 1.20 Photoshop.

Otro menú en común es el de **Vistas** (figuras 1.21 y 1.22). En los dos casos están las opciones de aumentar, reducir, encajar en pantalla, el trabajo con las guías, sectores y mostrar reglas. En GIMP existe, además, la posibilidad de una nueva vista, activar la ventana de navegación, mostrar los filtros y las barras de desplazamiento. En Photoshop, se pueden activar las correcciones de proporción de píxeles, colores de prueba, ajustes de prueba, avisar sobre gama, extras y bloquear o borrar sectores.

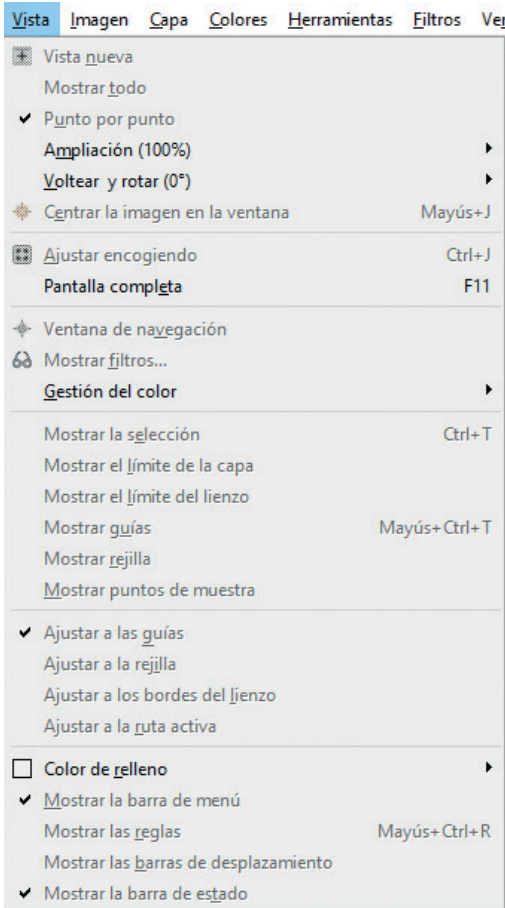


Fig. 1.21 GIMP.

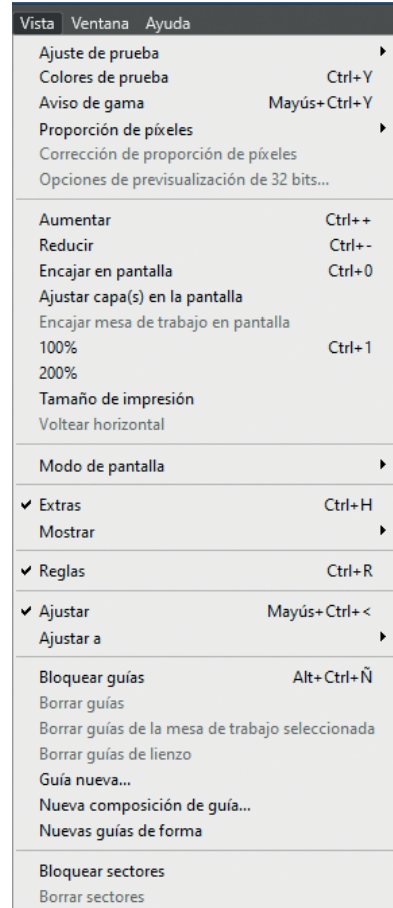


Fig. 1. 22 Photoshop.

El menú **Imagen (Image)** también coincide en ambas aplicaciones (figuras 1.23 y 1.24). En ambos casos aparecen las opciones de modo, transformación, duplicar; las de tamaño de imagen, recortar y separar. Pero en el caso de GIMP, se brindan además opciones para acoplar o alinear las capas visibles, guías; configurar la rejilla y las propiedades de la imagen. En Photoshop aparecen las opciones de calcular, la de proporción de píxel, rotar lienzo, separar y reventar.

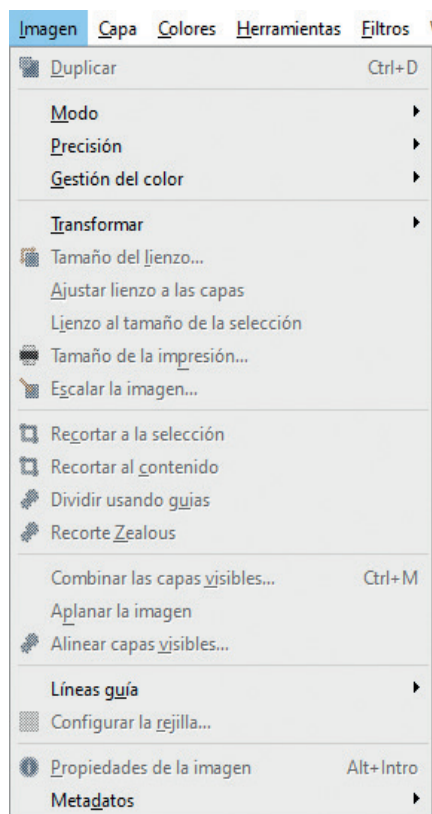


Fig. 1.23 GIMP.

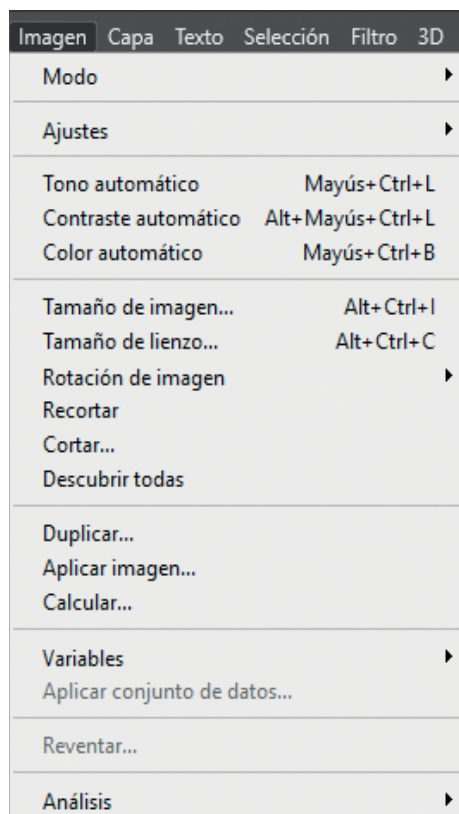


Fig. 1.24 Photoshop.

También el menú **Capas** se encuentra en ambas aplicaciones (figuras 1.25 y 1.26) y se caracteriza por tener las siguientes opciones: nueva capa, duplicar capa, eliminar capa, acoplar hacia abajo, estilo de capa, cambiar contenido de capa, máscaras, propiedades de capa y estilos de capa. Ahora bien, GIMP además permite trabajar con transparencia y cortar capas. En el caso de Photoshop, se puede crear un nuevo sector, crear máscaras de recorte, organizar, alinear, combinar visibles, crear nuevas capas de relleno y ajuste, cambiar contenido de capas, halos, acoplar imagen, texto y rasterizar.

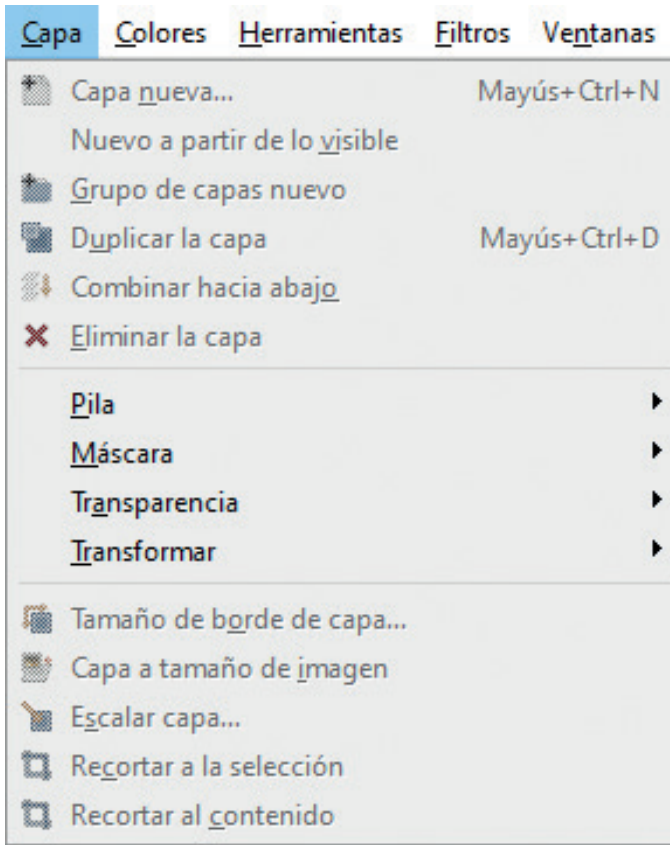


Fig. 1.25 GIMP.

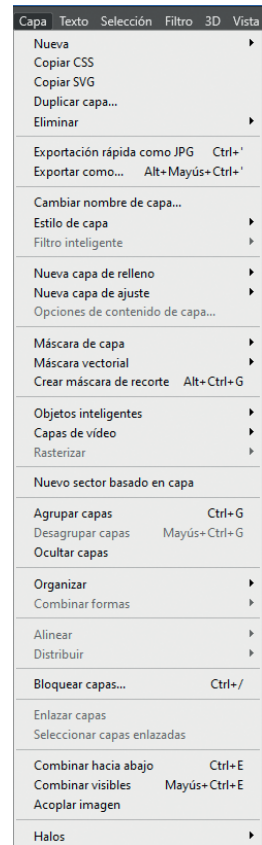


Fig. 1.26 Photoshop.

Otros de los menús comunes son el de **Filtros**, el de la **Ventana** y el de **Ayuda**. En el caso del menú **Ayuda**, en ambas aplicaciones se brinda el acceso a una serie de temas de ayuda con respecto a la aplicación, su funcionamiento y utilización. En el menú **Ventana**, en ambas aparecen opciones para determinar que ventanas se quieren mostrar o no en la aplicación. En el de **Filtros** aparecen las opciones de “último filtro”, seguidas por el nombre de todos los filtros que posee la aplicación en forma de lista, como se muestra en las imágenes que aparecen a continuación (figuras 1.27 y 1.28).

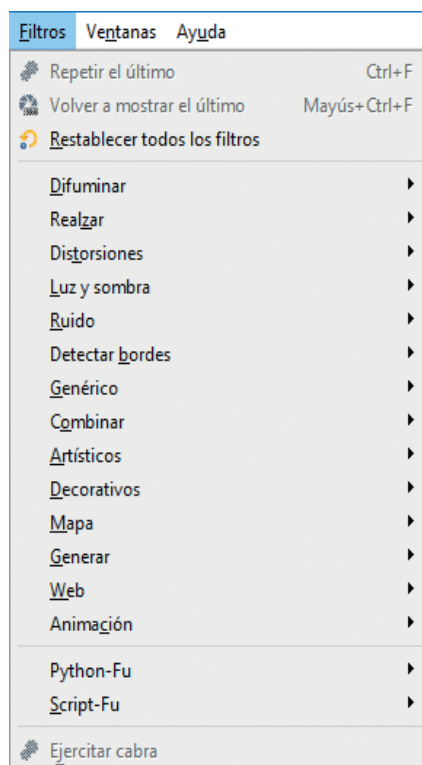


Fig. 1.27 GIMP.

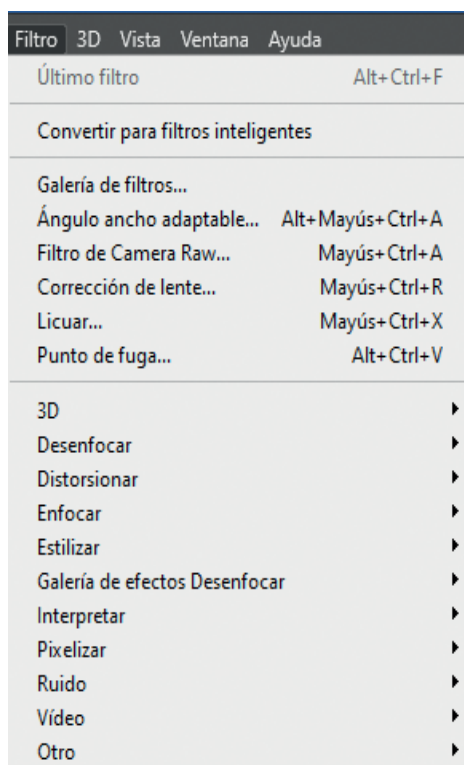


Fig. 1.28 Photoshop.

Entre las diferencias más explícitas se encuentran que en el menú **Color** de GIMP se brindan las opciones para el ajuste del balance de color, el tono, la saturación, brillo, contraste, niveles, curvas, saturación, entre otras relacionadas con el trabajo con las imágenes (figura 1.29) y en Photoshop estas opciones aparecen como parte del menú **Imagen**.

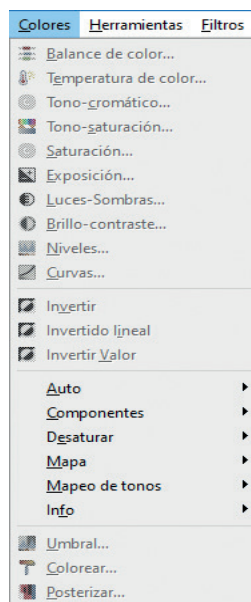


Fig. 1.29

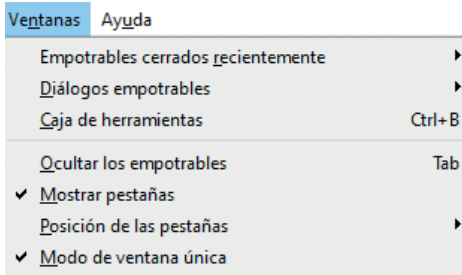


Fig. 1.30

También debes saber que en GIMP puedes agrupar y desagrupar los menús empleando Ventanas/ Modo de ventana única (figura 1.30).

Después de haber visto las diferentes herramientas y menús de ambas aplicaciones, pasaremos a lo que más nos interesa, el trabajo con la imagen, el cual se hará a partir de un grupo de prácticas guiadas que te permitirán aprender a realizar optimizaciones de imágenes, aplicar filtros y realizar fotomontajes en cualquiera de estas herramientas de edición de imagen digital. Siempre se mostrará el procedimiento paso a paso en las dos aplicaciones.

Práctica guiada 1

Imagina que estás creando una página Web y quieres insertarle una imagen ubicada en la carpeta "Mis imágenes" de "Mis documentos", pero te das cuenta que esta es muy grande, o sea, pesa mucho para su utilización en la red. Por tanto, necesitas reducir su tamaño para poder publicarla.

Lo primero que debes tener en cuenta son las **características** que afectan el tamaño del archivo de una imagen. En primer lugar, las **dimensiones de la imagen**: anchura y altura, expresadas en centímetros, milímetros, pulgadas o en píxeles. Luego la **resolución** de la imagen expresada, por ejemplo, en píxeles por pulgada (ppp), después la **profundidad de bit**, o sea, el número de niveles de gris o de colores diferentes que puede representar un píxel, pues a mayor profundidad de bit, mayor número de colores podrá representar un píxel y, por último, el **modo de color**, ya que de acuerdo con el **modo de color**, una imagen puede tener más o menos canales.

Cuanto mayor es el valor de cada una de las características expuestas, mayor será el tamaño del archivo de la imagen.

A continuación, se utiliza la figura 1.31 con el objetivo de mostrarte el procedimiento para trabajar estas características y disminuir el tamaño de la imagen.



Fig. 1.31

Observación

Siempre que vayas a trabajar una imagen, empieza por duplicarla antes de hacer cualquier cambio y utiliza esta copia para lograr el producto final, así mantendrás la original intacta y si algo saliera mal, siempre podrás disponer de la imagen original para comenzar de nuevo.

En GIMP

Comenzarás por abrir la imagen, para lo que tendrás que dar clic sobre el menú **Archivo (File)** y después en la opción **Abrir (Open)**. Luego en la ventana que aparece buscarás la ubicación de la imagen (**Places**) y darás clic en **Abrir** (figura 1.32).

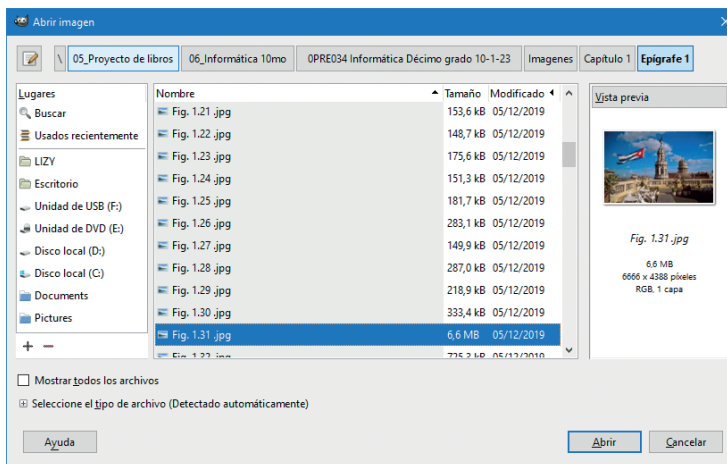
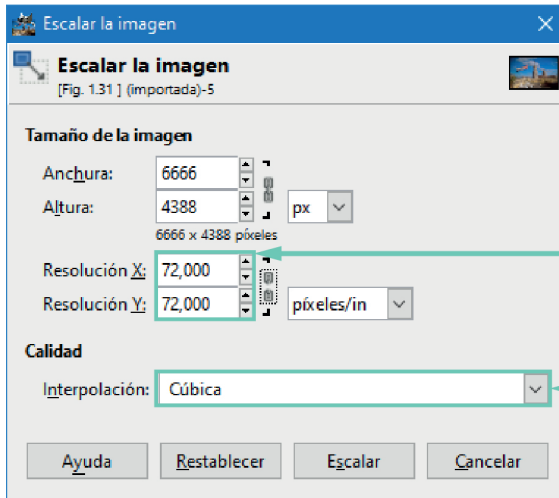


Fig. 1.32

Seguidamente en el menú **Imagen**, darás clic en **Duplicar** y en la ventana que aparece le darás nombre a la imagen, después clic en **Ok**. Luego cerrarás la original.

Con la copia de la imagen lista para trabajar, harás clic en el menú **Imagen (Image)**, en la opción **Escala de imagen (Scale Image)**. Entonces aparecerá la ventana donde podrás modificar el tamaño del documento, a partir del **ancho (Width)**, la **altura (Height)** y la **resolución** (figura 1.33).



Aquí estableces la resolución de la imagen.

Existen dos tipos de cambios de resolución: **remuestreo a la baja** y **remuestreo al alza**, que reducen o aumentan, respectivamente, la resolución de la imagen. Este proceso, se conoce como **interpolación**. Puedes seleccionar **ninguna**.

Fig. 1.33

Para ver cómo se trabaja en pulgadas o en centímetros, observa la figura 1.34, en la cual, al cambiar una de las dimensiones, la otra se ajusta automáticamente, también sigue las siguientes indicaciones:

1. Primero, en la figura se propone mantener la resolución en 72 ppp, porque como debes recordar esta es suficiente para la visualización con calidad de una imagen en el monitor de una computadora.
2. Después, se debe cambiar el ancho a 6 in (pulgadas), por lo que automáticamente el sistema ajustará la altura a 6,083 in, ya que se tiene la opción de **Restringir proporciones** de manera activada, de no ser así se podrían establecer las dimensiones según las necesidades. Luego, se podrán guardar los cambios dando clic en **Escalar (Scale)** (figura 1.35).

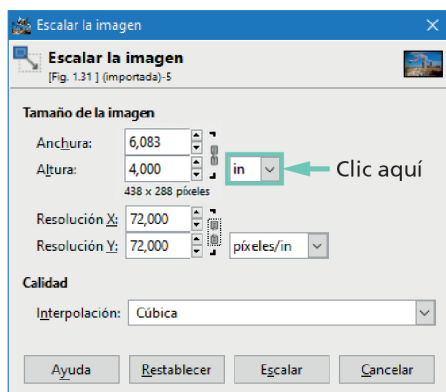


Fig. 1.34

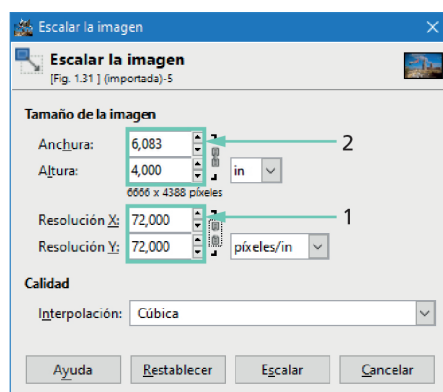


Fig. 1.35

También puedes volver a las opciones originales de la imagen dando clic en el botón **Resetear** (*Reset*) o cancelar los cambios en el botón **Cancelar** (*Cancel*).

Bueno, ya sabes cómo modificar las principales características de la imagen para optimizarla de acuerdo a tus necesidades, solo te falta guardarla en el formato más adecuado. Esto es simple, en el menú **Archivo** seleccionas la opción “Guardar como” y en la ventana que aparece establece (figura 1.36):

1. El lugar donde se va a guardar la imagen.
2. El nombre que tendrá la imagen.
3. El tipo de formato en el que se guardará.

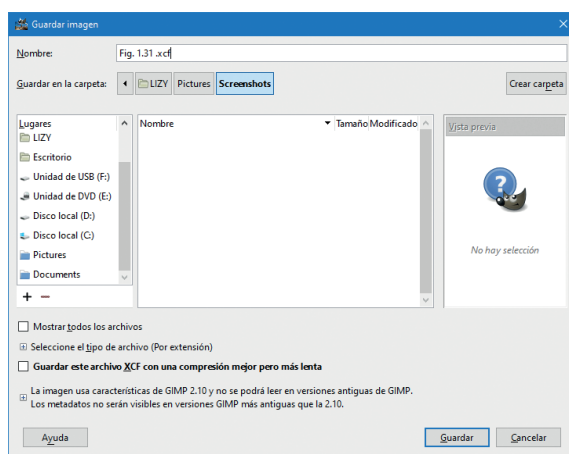


Fig. 1.36

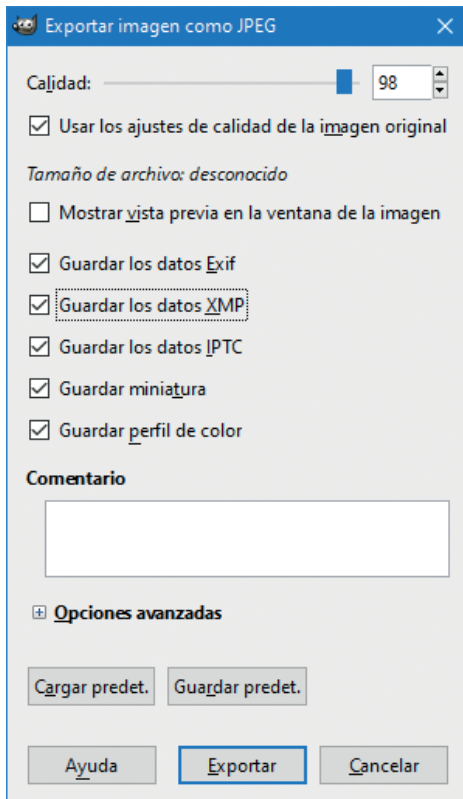


Fig. 1.37

Después, vas a desplegar las opciones dando clic en la saeta para ver los formatos. Si se elige JPG, se dará clic en el botón **Guardar (Save)** y aparecerá otra ventana que te permite elegir la cantidad de compresión que quieras realizar. En este caso, se ponen los valores que se muestran en la siguiente imagen y se hace clic en **Aceptar** (figura 1.37).

Ahora se te propone ver cómo se guarda en otros formatos. Siguiendo el mismo proceso descrito en el punto anterior, se seleccionará el formato de archivo BMP. En esta ocasión, debe aparecer la ventana de la figura 1.38 al hacer clic en **Guardar**. Después se hará clic en **Exportar** y nos aparece una nueva ventana como la de la siguiente imagen (figura 1.39).

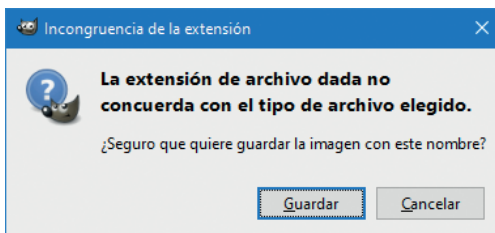


Fig. 1.38

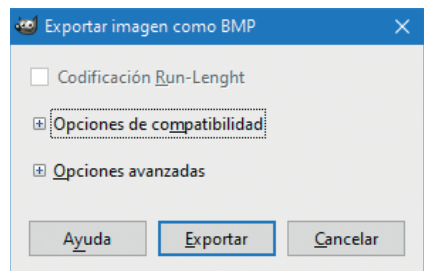


Fig. 1.39

Volverás a dar clic en **Guardar**. Se puede realizar el mismo proceso y elegir el formato de archivo TIFF. Observarás que al hacer clic en **Aceptar**, aparece una ventana similar a la que aparecía en el anterior formato.

Harás clic en **Exportar** y aparece una nueva ventana como la de la siguiente imagen (figura 1.40).

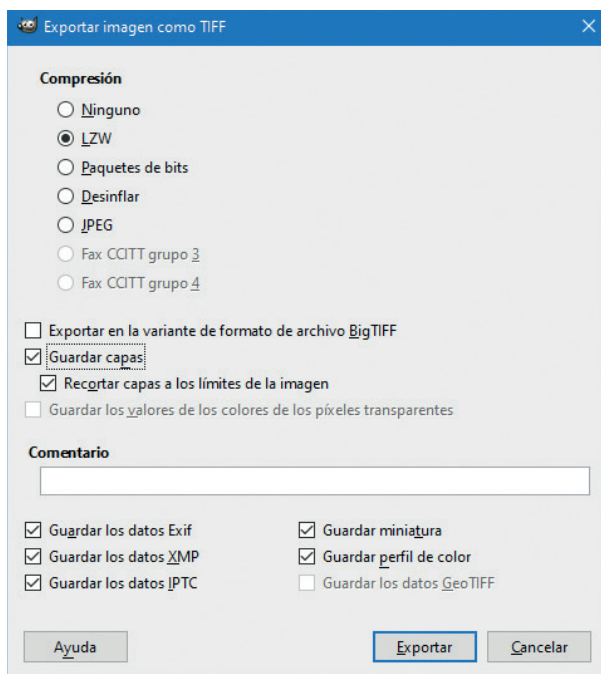


Fig. 1.40

En esta ocasión, se te permitirá guardar la imagen con algún tipo de compresión. Entonces elegirás una compresión sin pérdidas que hará que el archivo ocupe menos espacio en el disco, será LZW y harás clic en **Guardar**. Dado que la imagen lleva incrustado un comentario aparecerá una nueva ventana con un aviso sobre esta circunstancia. Darás clic en **Aceptar** para culminar el procedimiento.

En Photoshop

Comenzarás dando clic sobre el menú **Archivo**, allí tocarás la opción **Abrir**, luego en la ventana que aparece vas a buscar la ubicación de la imagen y darás clic en **Abrir**. Seguidamente en el menú **Imagen**, harás clic en **Duplicar** y en la ventana que aparece le pondrás nombre a la imagen y luego darás clic en **Ok**.

A continuación, cerrarás la imagen original. Con la copia de la imagen seleccionada para trabajar, darás clic en el menú **Imagen**, seguido de un clic en **Tamaño de imagen** (figura 1.41).

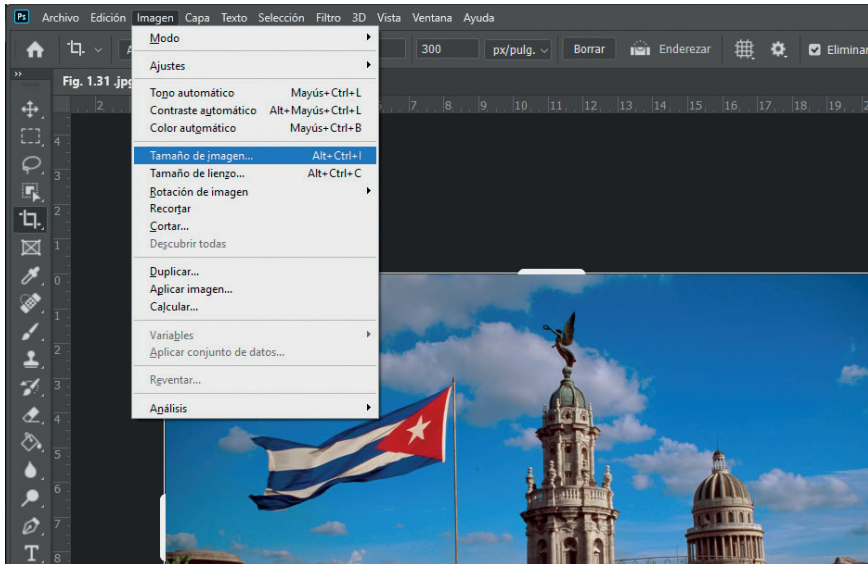
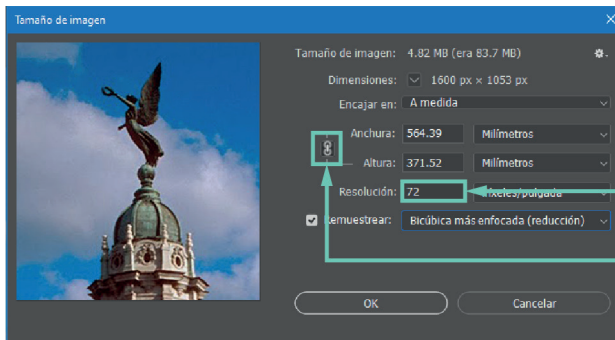


Fig. 1.41

Entonces, aparecerá la ventana donde podrás modificar el tamaño del documento a partir del ancho, la altura y la resolución. Analizarás algunos elementos importantes de esa ventana (figura 1.42).



Aquí estableces la resolución de la imagen.

Esta opción permite que al insertar un valor en el ancho o el alto de la imagen el sistema asigna automáticamente el valor adecuado para mantener una proporción adecuada.

Fig. 1.42

Luego, establecerás el tamaño y la resolución que se necesita para resolver esta problemática (figura 1.43):

1. Primero, debes mantener la resolución en 72 ppp, porque como debes recordar esta es suficiente para la visualización con calidad de una imagen en el monitor de una computadora.

- Después debes cambiar el ancho a 7 pulgadas, por lo que automáticamente el sistema ajustará la altura a 4,607 pulgada, ya que se tiene la opción de **Restringir proporciones** de manera activada, de no ser así, se podrían establecer las dimensiones según las necesidades (si te fijas en las dimensiones en píxeles, después de los cambios realizados, ha cambiado el tamaño del archivo de 4,82 Mb a 499,8 Kb. Luego podrás guardar los cambios dando clic en **Ok**. Si te equivocas puedes dar clic en el botón **Cancelar** y volverás a los parámetros originales de la imagen.

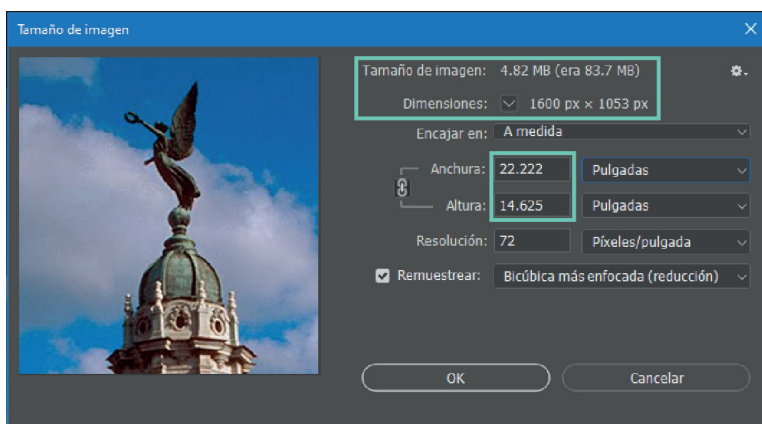


Fig. 1.43

Bueno, ya sabes cómo modificar las principales características de la imagen para optimizarla de acuerdo con tus necesidades, solo te falta guardarla en el formato más adecuado.

Esto es simple, en el menú **Archivo** seleccionas la opción **Guardar como** y en la ventana que aparece establecerás:

- El lugar donde se va a guardar la imagen.
- El nombre que tendrá la imagen.
- El tipo de formato en el que se guardará.

Para terminar, darás clic en el botón **Guardar** (figura 1.44).

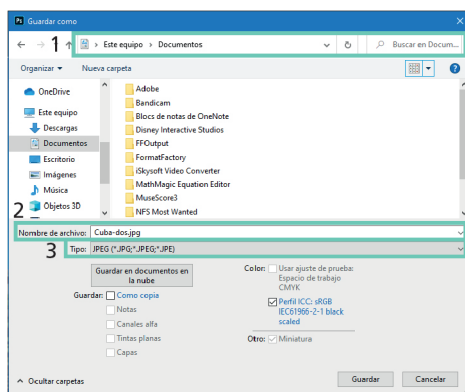


Fig. 1.44

Práctica guiada 2

Ahora imagina que tienes una imagen actual sobre los campos cubanos, pero quisieras tenerla en blanco y negro para que parezca que ha sido tomada muchos años antes. Entonces necesitarías cambiar el **modo de color**. A continuación, verás cómo se logra modificar los modos de color de una imagen digital.

Observación

Recuerda que siempre antes de ejecutar cualquier cambio debes comenzar por abrir la imagen original en **Archivo (File)/Abrir (Open)**, y luego crear la copia con la que vas a trabajar en **Imagen (Image)/Duplicar (Duplicate)**.

En GIMP

De manera general, para cambiar a cualquier modo de color que acepte el sistema, con la imagen lista para trabajar se iría al menú **Imagen (Image)/Modo(Mode)**, y allí se seleccionaría el modo de color deseado. En este caso sería **Escala de grises** (figura 1.45).

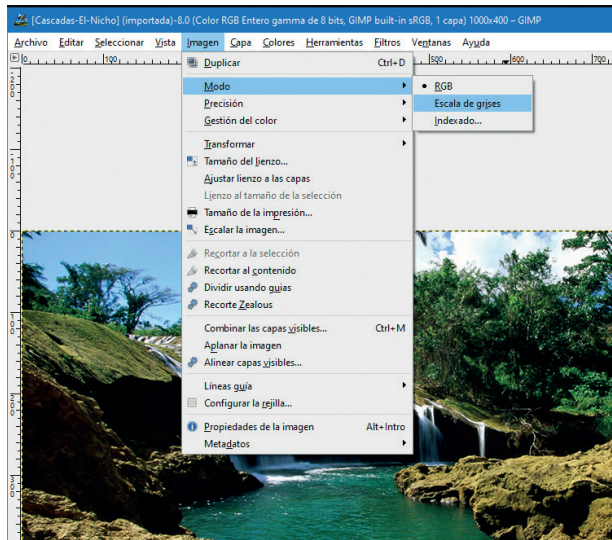


Fig. 1.45

Al dar clic sobre la opción **Escala de grises**, se obtiene como resultado la siguiente imagen (figura 1.46). Para terminar, la guardarás siguiendo el procedimiento que seguiste en la práctica guiada 1.

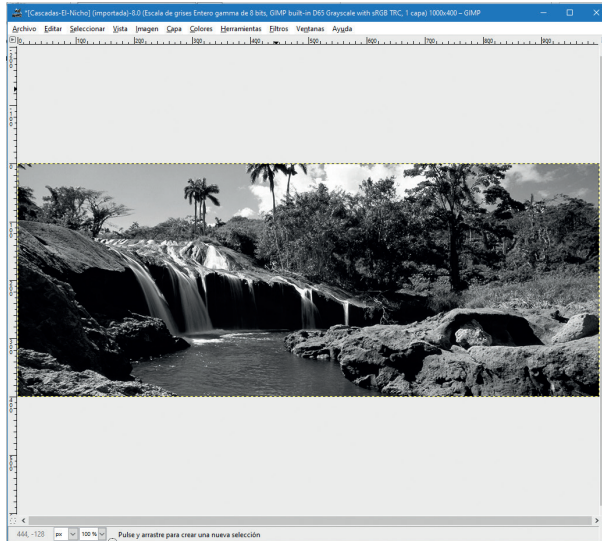


Fig. 1.46

En Photoshop

Realizarás la misma acción dando clic sobre el menú **Archivo/Abrir**, luego en la ventana que aparece buscarás la **ubicación de la imagen** y darás clic en **Abrir**. Seguidamente, en el menú **Imagen**, darás clic en **Duplicar** y en la ventana que aparece le pondrás nombre a la imagen y **luego** clic en **Ok**. Luego, cerrarás la original.

Para cambiar a cualquier modo de color que acepte el sistema, con la imagen lista para trabajar irás al menú **Imagen/Modo**, y seleccionarás el modo de color que desees utilizar. En este caso seleccionarás **Escala de grises** y aparecerá una ventana pequeña de confirmación (figura 1.47 a). Darás clic en **Ok** y obtendrás el siguiente resultado (figura 1.47 b).

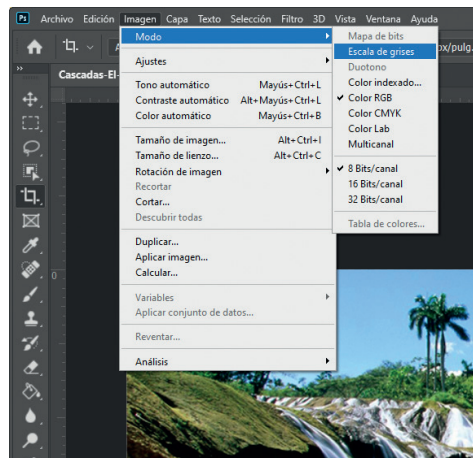


Fig. 1.47 a

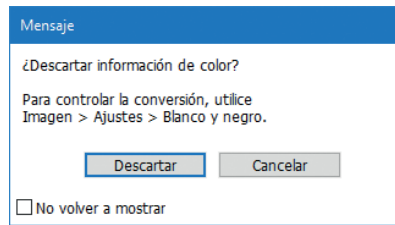


Fig. 1.47 b

Por último, guardarás la imagen en el formato que desees, como se vio en otras prácticas anteriores (figura 1.48).

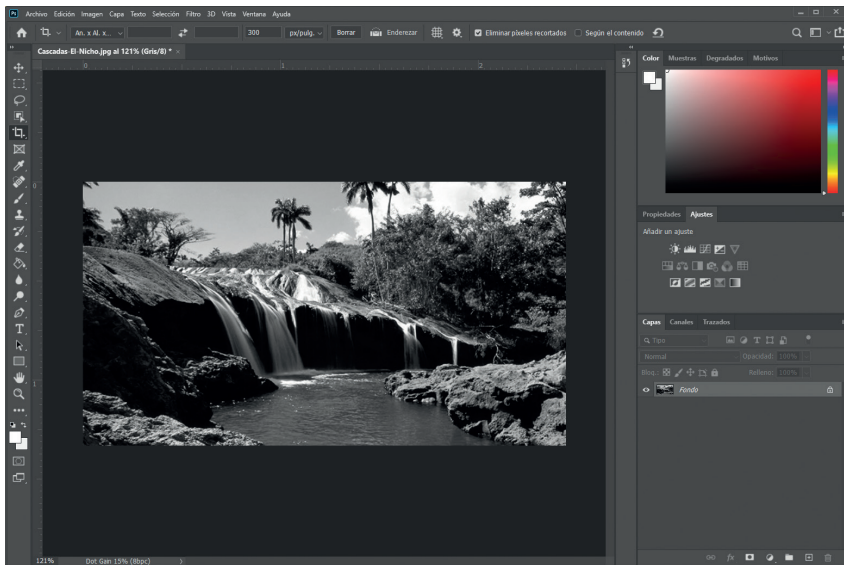


Fig. 1.48

Se pasará ahora a las opciones de retoque y transformación.

Práctica guiada 3

Supón que has escaneado una foto de un libro de historia que necesitas para un trabajo práctico, pero como el libro es muy viejo, quisieras retocarla para quitar las manchas u otros defectos que aparecen por el paso del tiempo. La foto sería la siguiente (figura 1.49).



Fig. 1.49

Como puedes observar está inclinada, posee un fondo negro y algunas imperfecciones, además el color es muy opaco. Ahora, veremos cómo solucionar esta problemática, dejar la foto sin imperfecciones y ya lista para imprimir y ser utilizada en nuestro trabajo práctico.

En GIMP

Recuerda comenzar por crear la copia con la que trabajarás en **Imagen (Image)/Duplicar (Duplicate)**. Una vez con la imagen lista comenzarás por rotarla para lograr que quede recta. GIMP tiene en el menú **Imagen (Image)** (figura 1.50 a) la opción **Transformar (Transform)**, que te permitirá rotar una imagen en ángulos de 90° o 180° y voltearla vertical u horizontalmente a partir de ajustes manuales (**Gillotine**).

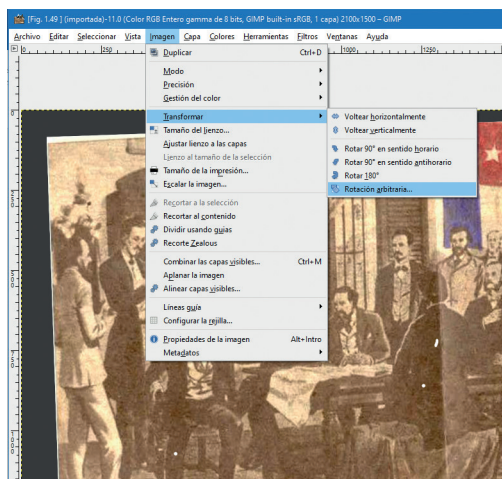


Fig. 1.50 a



Fig. 1.50 b

También en la barra de herramienta hay una para este fin, que es la que aparece seleccionada en la figura 1.50 b.

Por cualquiera de las dos vías, darás clic sobre el área que desees rotar y en la ventana que aparece desplazarás el botón deslizante hasta conseguir el ángulo deseado. Deberás dar clic en el botón **Rotar** para hacer efectivo el cambio. Se puede regresar al estado original dando clic en el botón **Resetear (reset)** o puedes cancelar la acción en el botón **Cancelar** (figura 1.51).

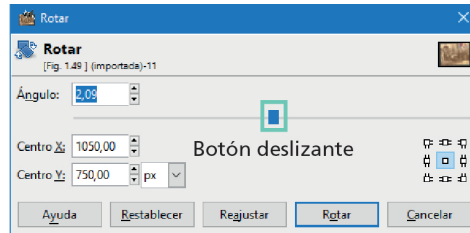


Fig. 1.51

Luego, pasarás a eliminar el fondo de color negro. Empezarás seleccionándolo, puedes utilizar cualquiera de las herramientas de selección (recuerda que cada una tiene usos específicos). En este caso, utilizarás la herramienta  que te permite hacer selecciones continuas de un mismo color (figura 1.52).



Fig. 1.52

Poco a poco, te irás situando en un lugar de la superficie de color negro y allí darás clic. Inmediatamente, aparecerá una selección alrededor de la imagen, presionarás la tecla **Delete** y todo lo que está seleccionado se borrará (figura 1.53).

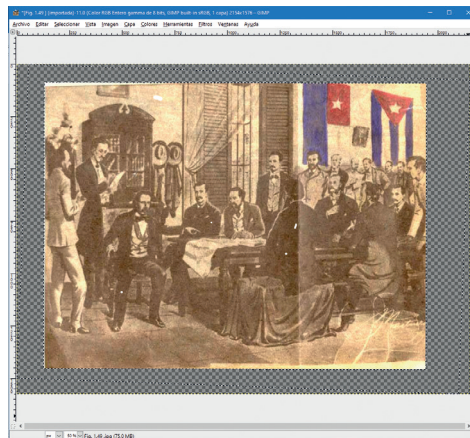


Fig. 1.53

Observación

En general, cuando se crea una selección, se visualiza como una línea de guiones que encierra una porción de la imagen. La idea es que la **selección** es una suerte de contenedor, con las partes de la imagen seleccionadas adentro, y las partes no seleccionadas afuera.

Pasarás después a corregir las imperfecciones, para eso usarás la herramienta de **Curación** ✂. Esta se utiliza presionando la tecla "Ctrl + clic en el lugar" a partir del cual vas a corregir la imperfección, y luego con el botón primario del ratón presionado lo arrastras sobre la imperfección hasta cubrirla. Te dará la siguiente forma (figura 1.54 a, b). A continuación, realizarás los ajustes de los colores y lo harás seleccionando la opción deseada en el menú **Colores (Colors)**. Comenzarás ajustando el **Balance del color (Color balance)**. Al concluir darás clic en **Ok**.



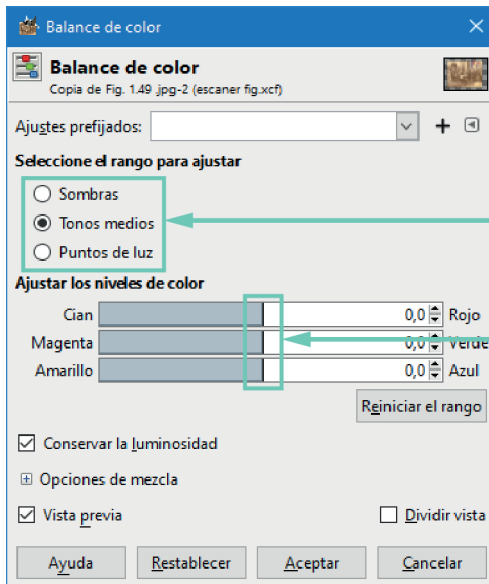
a



b

Fig. 1.54

Luego trabajarás sobre el brillo y el contraste, en estos aspectos podrás hacer diferentes cambios al **tono (Hue)** y la **saturación (Saturation)** utilizando los **niveles (Levels)**, las **curvas (Curves)**, **desaturar (Desaturate)**, entre otros cambios que te permite hacer la herramienta, según tus necesidades (figura 1.55).



Aquí se selecciona a qué se le hará el ajuste, a las sombras (*Shadows*), a los medios tonos (*Midtones*) o a las iluminaciones (*Highlights*)

Aquí se desliza el botón y se selecciona a qué gama de colores uno se acerca.

Fig. 1.55

Procederás de la misma forma en la mayoría de las ventanas, a partir de la selección, mediante el botón deslizante para darle más o menos brillo y más o menos contraste a la imagen. Esta ventana también te permite ajustar estos elementos a partir de los niveles (figura 1.56).

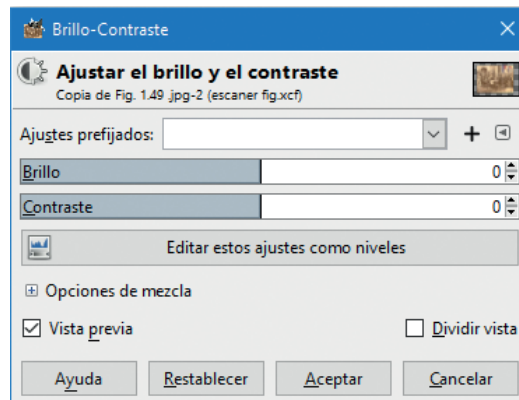
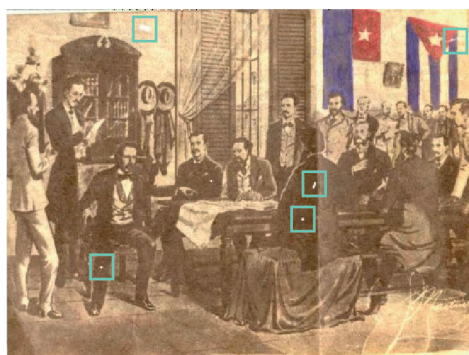


Fig. 1.56

Aquí tienes la imagen inicial, obsérvala bien y verás los cambios (figura 1.57 a), compárala con la imagen final (figura 1.57 b).



a



b

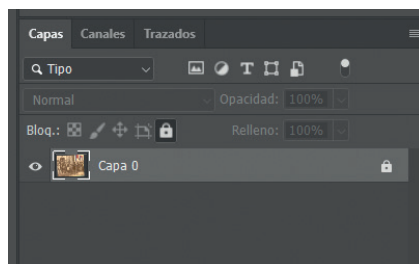
Fig. 1.57

En Photoshop

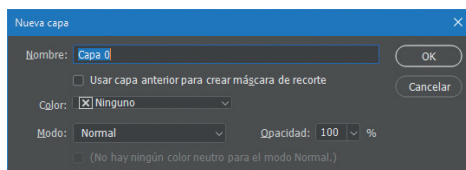
Siempre debes comenzar por crear la copia con la que trabajarás en el menú **Imagen/Duplicar**. Ya con la imagen abierta, lo primero es desbloquear la capa para poder trabajar con ella y luego, entonces, cambiar su inclinación.

Para desbloquear una capa, deberás ir a la ventana de **Capas** y dar doble clic en la capa que está bloqueada (la puedes identificar porque aparece un candado pequeño junto al nombre de la capa). Luego, en la ventana que aparece le pondrás nombre a la capa y seleccionará, si lo deseas, un color de fondo, además podrás determinar el modo y la opacidad (figura 1.58 a y b).

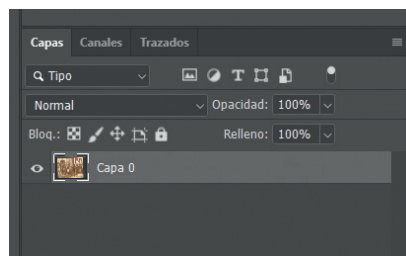
Seguidamente, ya la capa estará desbloqueada y podrás comenzar el trabajo con la imagen (figura 1.58 c). Para rotarla irás al menú **Edición/Transformar/Rotar** (figura 1.59).



a



b



c

Fig. 1.58

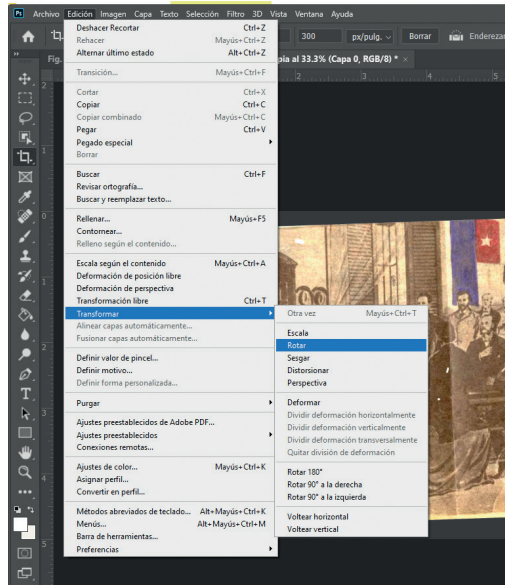


Fig. 1.59

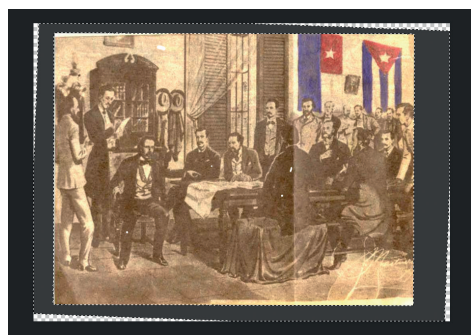
A continuación, por el borde de la imagen aparecerán unas líneas discontinuas y unas guías para que al dar clic sobre ellas puedas girar la imagen hasta la posición deseada. Después de tener la imagen en el lugar que desees, darás doble clic y ya tendrás el cambio hecho (figura 1.60).



Fig. 1.60

Luego pasarás a seleccionar el fondo para eliminarlo (figura 1.61 a), al ser un color entero utilizarás la herramienta **Varita mágica** . Darás clic sobre cualquier lugar del fondo y presionarás la tecla **Delete** y quedará eliminado el fondo. Posteriormente, usarás la herramienta **Recortar**  y eliminarás todo el fondo que te sobra, dando clic y arrastrando hasta obtener el tamaño deseado.

Concluirás esta actividad dando doble clic sobre la imagen, la cual queda recortada (figura 1.61 b).



a



b

Fig. 1.61

A continuación, pasarás a corregir las imperfecciones. Con la herramienta **Lupa** ampliarás la imagen para trabajar con mayor precisión y luego seleccionarás la herramienta **Pincel corrector**. Seguidamente, presionarás la tecla "Alt + clic sobre el lugar" que vas a tomar como referencia para corregir la imperfección, y después la arrastrarás sobre la imperfección y obtendrás los siguientes resultados (figura 1.62 a, b).



a



b

Fig. 1.62

Repetirás este procedimiento tantas veces como sea necesario para corregir todas las imperfecciones, luego, para darle mejor terminación en cuanto al color, seleccionarás el menú **Imagen/Ajustes/Equilibrio de color**, al igual que en GIMP los botones deslizantes se ajustarán según el gusto del operador y luego se dará clic en **Aceptar** (figura 1.63). A continuación, se realizará el ajuste en el brillo y el contraste, con este fin se utilizarán también los botones deslizantes (figura 1.64).

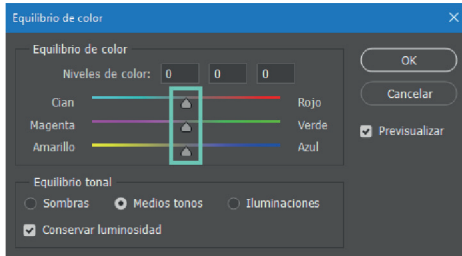


Fig. 1.63

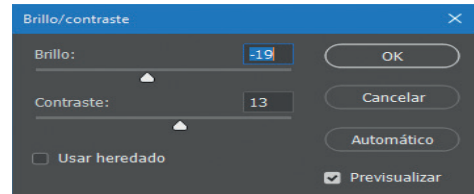


Fig. 1.64

Se culminará el procesamiento de esta imagen, guardándola a partir de la opción "Guardar como" del menú **Archivo**. En esta ventana buscarás la ubicación donde la vas a guardar, establecerás el nombre de la imagen y el formato con el cual la guardarás (figura 1.65).

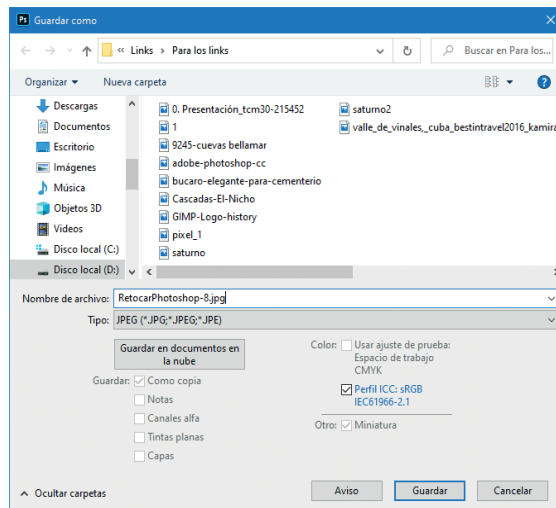


Fig. 1.65

Seguramente, aparecerá la ventana de las opciones del formato en el que deseas guardar la imagen, realizarás las modificaciones que consideres necesarias y darás clic en **Ok** para concluir el procedimiento (figura 1.66).

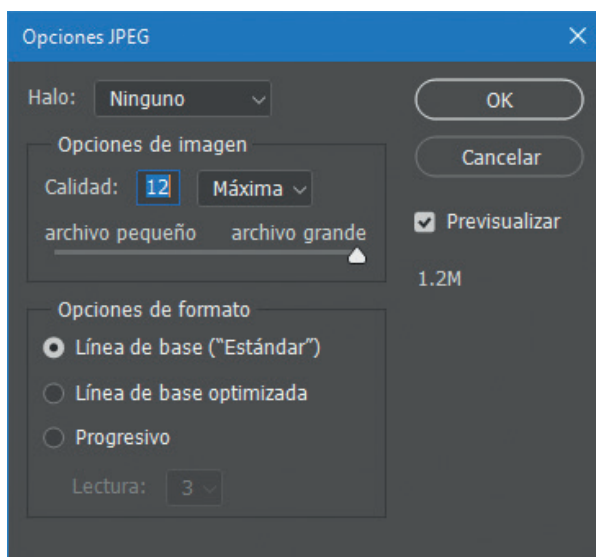


Fig. 1.66

A continuación, se te propone observar estas imágenes para que evidencies los cambios realizados (figura 1.67 a, b).



a. Imagen original.



b. Imagen retocada.

Fig. 1.67

Práctica guiada 4

Como parte de una investigación para la jornada científica de Ciencias Naturales, quieres mostrar una infografía donde aparezcan reflejados fenómenos como la desertificación de los suelos, la sequía y la contaminación de las aguas. Para esto, lo primero que tienes que hacer es seleccionar las imágenes que vas a utilizar. En este caso serán las que siguen a continuación (figura 1.68 a, b, c, d).



a



b



c



d

Fig. 1.68

En GIMP

Empezarás por crear una nueva imagen en GIMP que será la base del fotomontaje que realizarás y dará lugar a la infografía. Lo harás dando clic en el menú **Archivo (File)**, donde seleccionarás la opción **Nuevo (New)** y verás una ventana.

En ella aparecerá la opción de **Utilizar una plantilla (Template)** dando clic sobre la saeta, pero de no desear esto puedes hacerlo manualmente estableciendo ancho (**Width**) o alto (**Height**) en píxel, pulgada o las unidades de medidas que ofrece el sistema. De querer modificar también la resolución (**Resolution**), el modo de color (**Color space**), tipo de fondo (**Fill with**), deberás dar clic sobre la opción **Opciones avanzadas (Advanced options)** (figura 1.69).

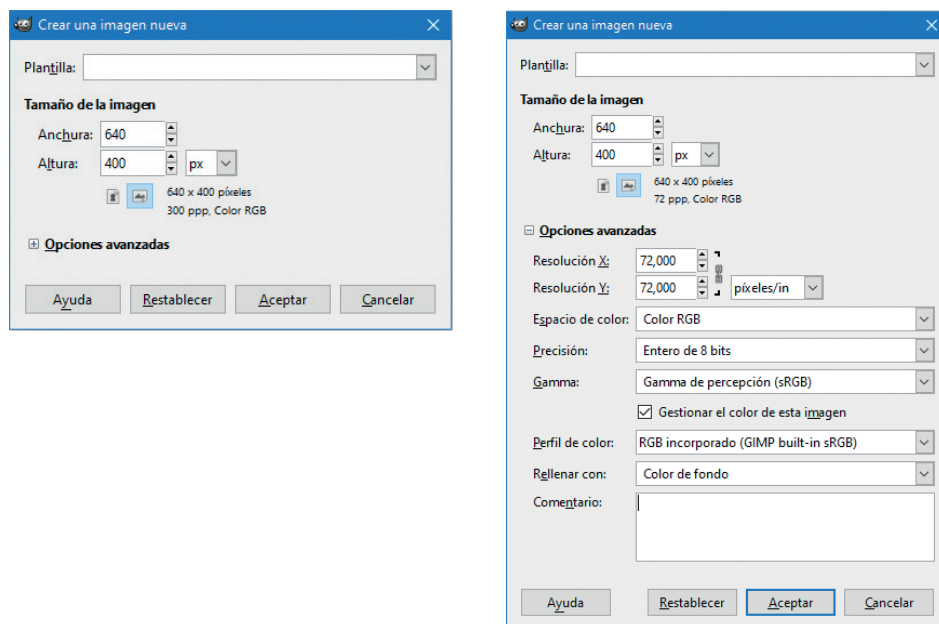


Fig. 1.69

En este caso, lo harás estableciendo una resolución de 300 ppp, ya que la infografía será para imprimir con 8 pulgadas de ancho por 10 pulgadas de altura, con un modo de color RGB y un color de fondo blanco. Cuando ya tengas la base para trabajar, que de forma predefinida se llama "Sin título 1" (*Untitled 1*), entonces abrirás el Explorador de Windows y buscarás la ubicación de las imágenes, pondrás la vista en miniaturas y arrastrarás una a una estas imágenes hacia la base en blanco creada en GIMP. Como podrás apreciar, cada imagen se agrega como una nueva capa. Te quedará como muestra la imagen que aparece a continuación (figura 1.70).

En esta ventana de capas, seleccionarás la activa dando clic sobre ella, luego realizarás algunas acciones con ella, como:

1. Eliminarla.
2. Crear o duplicarla.
3. Moverla hacia arriba o abajo (esto significa que se muestre delante o detrás de las capas que la anteceden o la preceden).

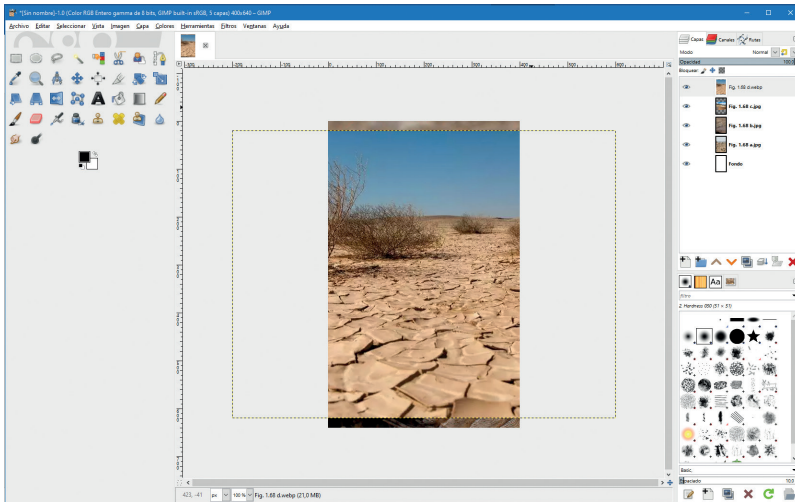


Fig. 1.70

También podrás crear una nueva en la primera opción de la ventana. Posteriormente, ajustarás los tamaños de cada una de esas capas. Comenzarás por dar clic sobre la capa donde está la imagen que se quiere modificar y luego en el menú **Capas (Layer)** seleccionarás la opción **Escala de capa (Scale Layer)**.

En la ventana que aparece, establecerás las dimensiones que desees para la capa, en este caso elegirás en pulgadas. Por último, darás clic en la opción **Escalar** (figura 1.71). Este procedimiento lo repetirás tantas veces como capas tengas que redimensionar.

El resultado sería el que se muestra en la figura 1.72.

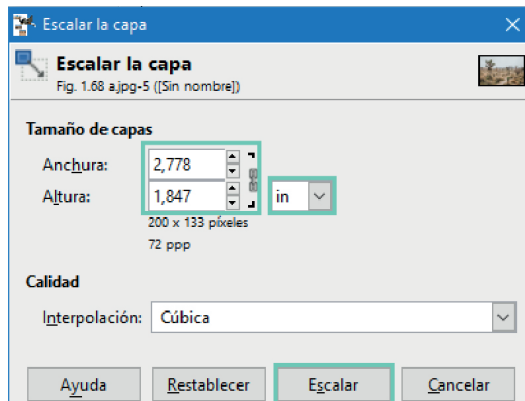


Fig. 1.71

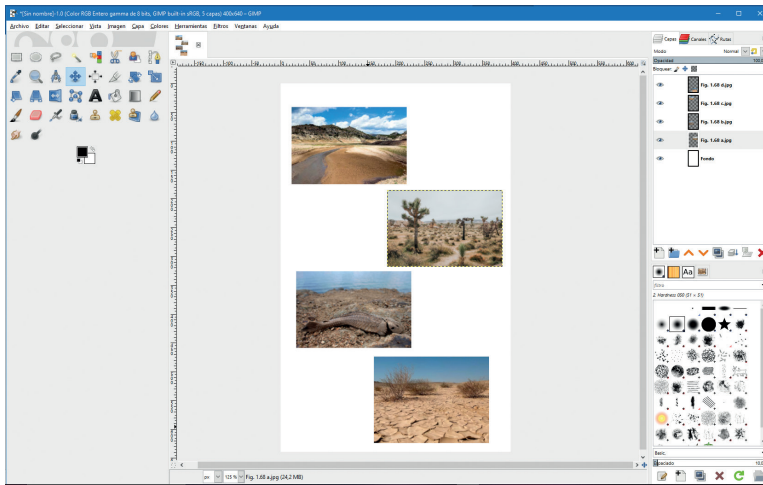


Fig. 1.72

Ahora bien, es momento de organizar las imágenes, para esto utilizarás la herramienta de Selección , y en la ventana de **Capas** las organizarás utilizando las opciones   para poner delante o detrás las imágenes, de acuerdo a tu gusto o necesidad. En este caso queda de la siguiente forma (figura 1.73). Pero estas imágenes por sí solas no te dirán nada, por lo que habrás de agregarle un texto. Para agregar el texto en la ventana de herramientas, seleccionarás la herramienta  y a continuación aparecerá la ventana que se muestra en la figura 1.74.

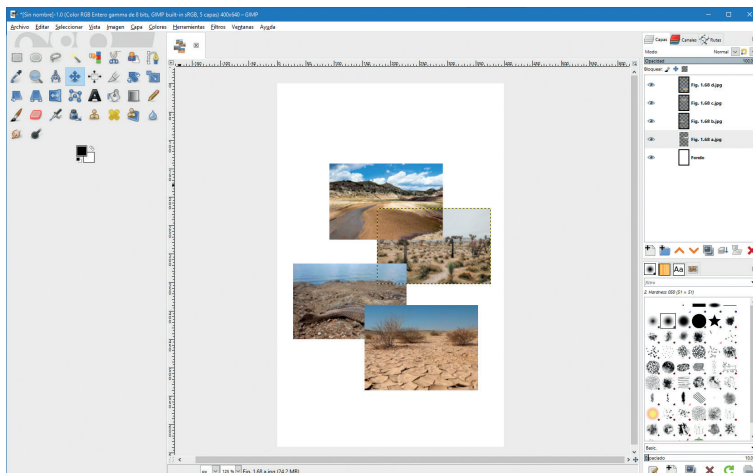


Fig. 1.73



Fig. 1.74

Donde aparece el cursor introducirás el texto deseado y luego presionarás el botón **Cerrar**. Obtendrás el siguiente resultado (figura 1.75).

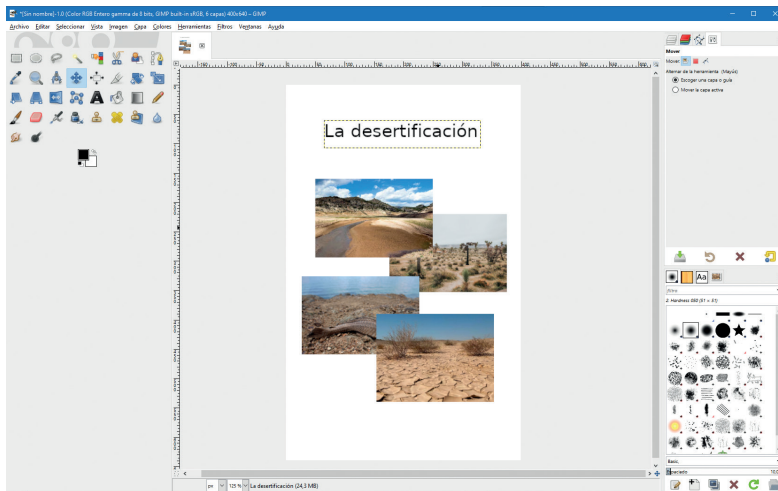


Fig. 1.75

Repetirás estas acciones tantas veces como textos independientes quieras insertar. Posteriormente, pasarás a darle un poco de color a la imagen, aplicando un fondo.

Primero seleccionarás la herramienta , que permite rellenar espacios de un mismo color. Al seleccionarla, aparecerán debajo de la ventana de herramientas sus opciones. Donde podrás definir:

- El modo (**mode**) de relleno y la opacidad (**opacity**) de este.
- El tipo de relleno (figura 1.76).
- A quién afectará y otras características sobre los colores similares.

En este caso, usarás “en modo normal”, la opacidad se mantendrá en 100, pero como tipo de relleno utilizarás un *Pattern fill*. Harás clic en el botón de radio y luego sobre la pequeña muestra podrás seleccionar el patrón deseado.

Después te situarás sobre el fondo blanco y darás clic, entonces aparecerá todo el fondo con el patrón seleccionado como se muestra en la imagen de la figura 1.77.

Ya solo te falta darle un poco de efecto al fondo. Con este fin, aplicarás un filtro. Irás al menú **Filtro (Filters)** donde aparecerán todos los filtros que puedes aplicar y seleccionarás el filtro que desees. En este caso, harás la selección dentro de los filtros artísticos (**Artistic**), el nombrado **Tramar** (figura 1.78).

Inmediatamente aparece esta ventana, donde a partir de los botones deslizantes decidirás las características del filtro que estas aplicando. Al concluir los ajustes, darás clic en **Ok** para terminar el procedimiento y obtener el siguiente resultado (figura 1.79 a, b).

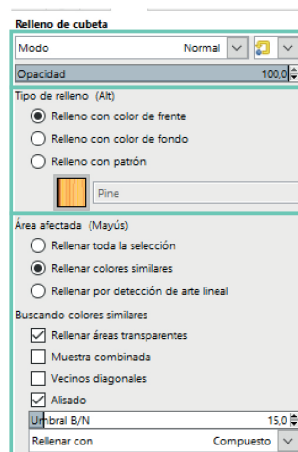


Fig. 1.76

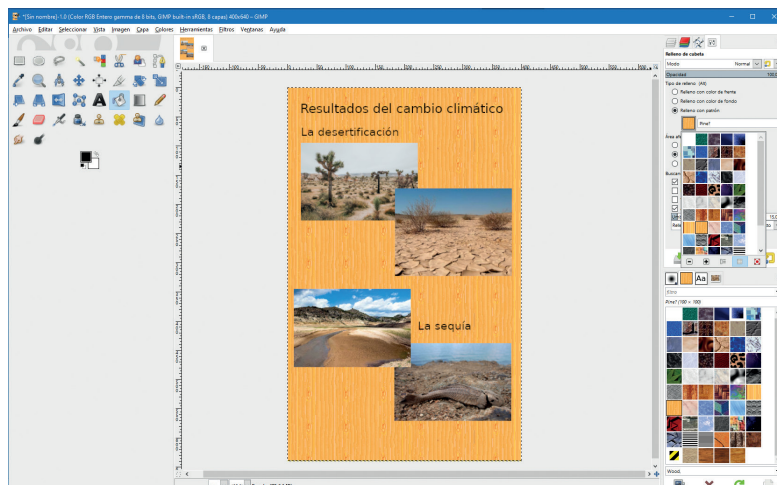


Fig. 1.77

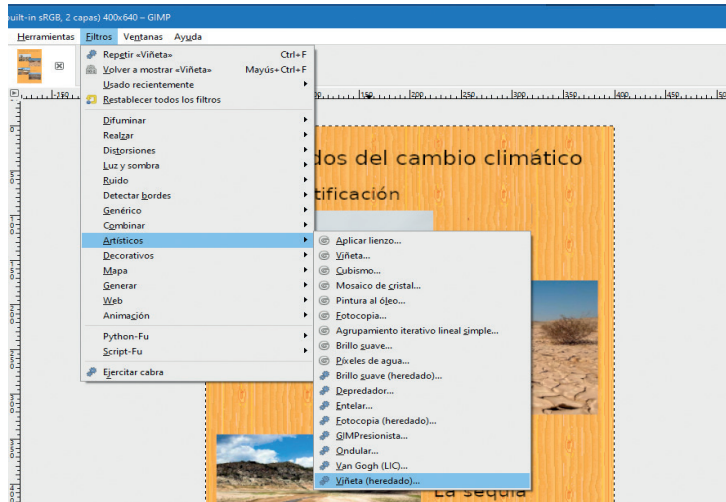
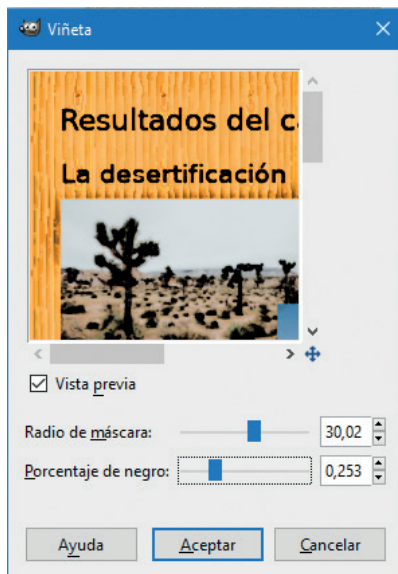
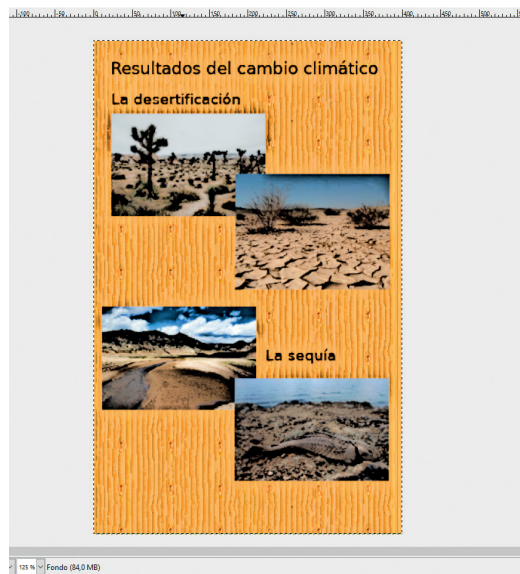


Fig. 1.78



a



b

Fig. 1.79

En Photoshop

Comenzarás por abrir las cuatro imágenes con las que vas a trabajar a partir del menú **Archivo/Abrir**, las seleccionarás.

Con estas imágenes abiertas crearás una nueva que va a ser la base de tu montaje, darás clic en la opción **Nuevo** del menú **Archivo**.

En la ventana que aparece especificarás las dimensiones de la imagen, la resolución, modo de color y color de fondo, luego harás clic en **Ok** y obtendrás la nueva imagen (figura 1.80).

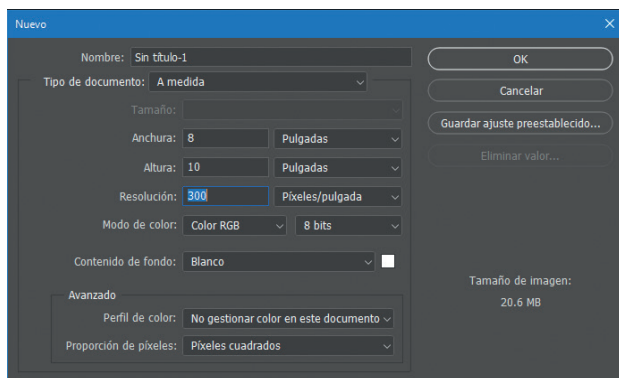


Fig. 1.80

Después pasarás a incorporar en esta nueva imagen las que abriste anteriormente (figura 1.81). Esto lo harás seleccionando la herramienta Mover , y arrastrarás cada una de las imágenes hacia el espacio de trabajo del nuevo documento creado, el que quedará así (figura 1.82), donde cada foto es una nueva capa de la imagen.

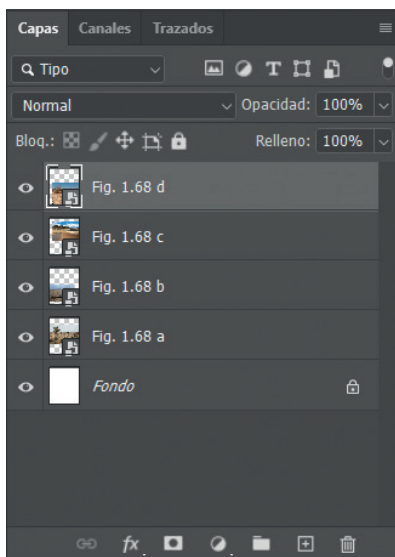


Fig. 1.81



Fig. 1.82

Como te puedes dar cuenta, por su tamaño, han quedado unas sobre otras y no se pueden ver bien, por lo tanto comenzarás por cambiar su tamaño. Esto lo podrás hacer seleccionando primeramente la capa de la imagen con la que vas a trabajar, dando clic sobre la miniatura de la capa. La capa seleccionada siempre aparece resaltada (figura 1.83).

A continuación, en el menú **Edición/Transformar** seleccionarás la opción **Escala** y allí ajustarás el tamaño como se hizo en la **práctica guiada 3**. Quedará de esta forma (figura 1.84).

Con la herramienta **Mover** ubicarás las imágenes de la forma que más te guste (figura 1.85).

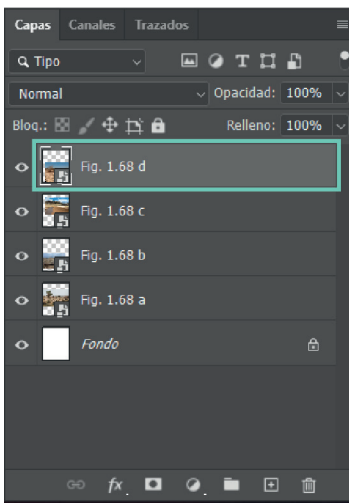


Fig. 1.83



Fig. 1.84



Fig. 1.85

Luego pasarás a escribir el texto que te permitirá decir que tienes en la imagen. Para esto seleccionarás la herramienta **Texto**  y comenzarás a escribir lo que desees. En la barra que aparece en la parte superior, después de seleccionar el texto, podrás cambiarle el color, el tamaño, el tipo de letra y otras opciones (figura 1.86).

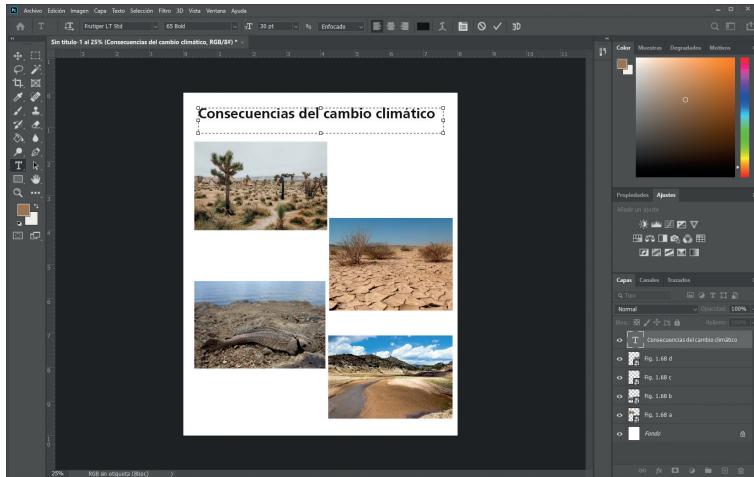


Fig. 1.86

A continuación, pasarás a la acción de modificar el fondo, al igual que en GIMP, utilizarás la herramienta **Bote de pintura** y en las opciones que aparecen en la barra superior seleccionarás **Rellenar con motivo**, y elegirás el motivo (figura 1.87). Bueno, el resultado sería el siguiente (figura 1.88).

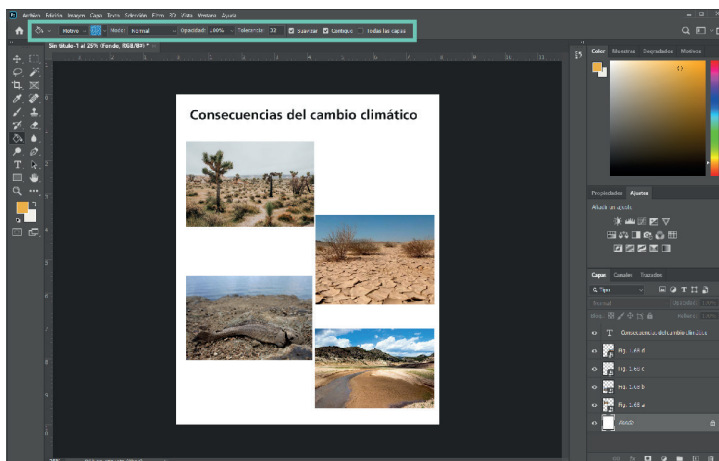


Fig. 1.87

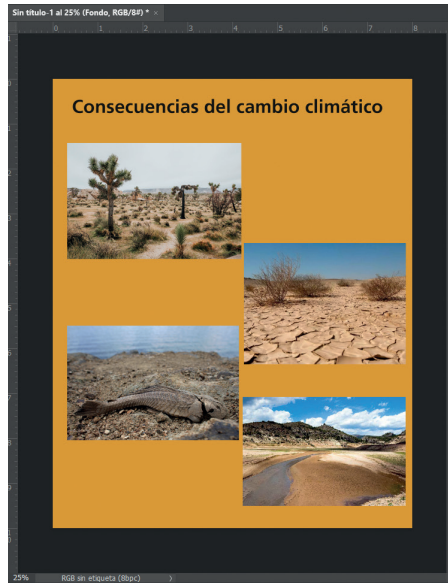


Fig. 1.88

Para finalizar, aplicarás un filtro al fondo para darle un mejor acabado. Para ello seleccionarás el menú **Filtro**, luego **Pixelizar** y después **Cristalizar** (figura 1.89). Más adelante seleccionarás, a partir del botón desplazable, el tamaño de cada celda y darás clic en **Ok** (figura 1.90). Ahora ya tienes listo el fotomontaje, que se verá así (figura 1.91).

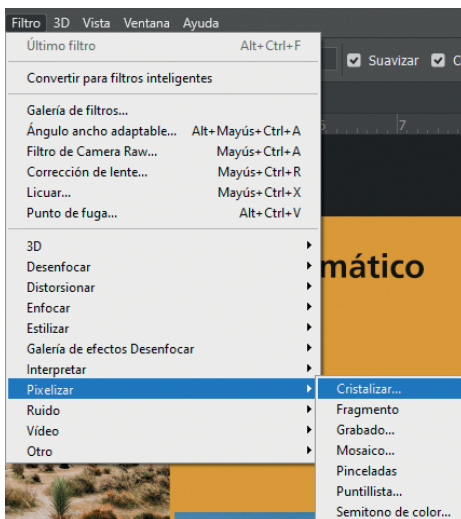


Fig. 1.89

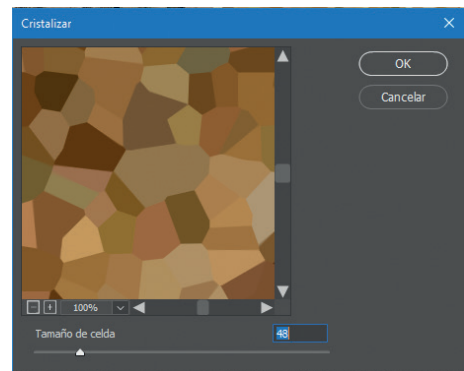


Fig. 1.90



Fig. 1.91

Comprueba lo aprendido

1. Realiza un resumen donde des respuesta a las siguientes interrogantes:
 - ¿Cómo pueden ser las imágenes digitales? ¿En qué se diferencian?
 - ¿Cuáles son los principales elementos que caracterizan a las imágenes digitales de mapa de bits?
 - ¿Qué es la resolución de una imagen de mapa de bits? ¿En qué consiste el fenómeno de pixelación?
2. Elabora un mapa conceptual donde establezcas las relaciones existentes entre los siguientes elementos:
 - Imagen digital
 - Imagen de mapa de bits
 - Resolución

- Imagen vectorial
- Profundidad de color
- Formato
- Modo de color
- Cámaras fotográficas digitales, escáner y capturas de pantalla

3. ¿Qué indica el modo de color en una imagen de mapa de bits? Indica los principales modos de color y las características básicas de cada uno.

4. Completa en la siguiente tabla los valores que faltan.

Profundidad de color	Cantidad de Colores
1 bit	2
4 bits	16
8 bits	
16 bits	
32 bits	

5. Tenemos una imagen de 12 bits. ¿De cuántos bits puede disponer la paleta de colores como máximo?

6. En el modo RGB, ¿qué color se representa con la notación (0,0,0)? ¿Por qué?

7. Completa los espacios en blanco con las palabras que aparecen a continuación, según corresponda:

formato

imagen

pantalla

a) Un _____ gráfico define la manera en que se guarda una imagen y toda la información relacionada con esta.

b) La resolución de la _____ se refiere al número de píxel por pulgada que es capaz de mostrar.

c) La resolución de una _____ es el grado de detalle o calidad de una imagen digital, ya sea escaneada, fotografiada o impresa.

8. Sabiendo que una fotografía tiene como medidas físicas 10 cm x 15 cm y su tamaño es de 2 271 x 1 513 píxeles. Calcula la resolución en ppp.



Recuerda que...

La resolución de una imagen se obtiene multiplicando su ancho por su alto, y 1 pulgada = 2,54 cm.

9. Si el archivo de una imagen tiene unas dimensiones de 5,184 x 3,472 píxeles, ¿de cuántos megapíxeles es la cámara de fotos?
10. Calcula el tamaño máximo de impresión (en centímetros) de una fotografía que tiene una resolución de impresión de 200 ppp y una calidad máxima de 5,184 x 3,456 píxeles.
11. Calcula que resolución en píxeles por pulgada tendrá una imagen cuyas características son las siguientes:
- Dimensiones en píxeles: 3,264 x 2,448
 - Longitud en centímetros: 115,15 x 86,36
12. Marca con la letra del inciso el concepto con su definición:

A

B

- | | |
|-------------------------|---|
| a) Píxel | ___ Es el sistema de coordenadas que permite describir el color de cada píxel utilizando valores numéricos. |
| b) Profundidad de color | ___ Es el número de bits necesarios para codificar y guardar la información de color de cada píxel en una imagen. |
| c) Modo de color | ___ Son las unidades de color que componen la imagen digital de mapa de bits. |

d) Resolución

__ Define la manera en que se guarda una imagen y toda la información relacionada con esta.

e) Formato gráfico

__ Define el tamaño de una imagen, describe sus dimensiones físicas y se mide en píxel.

13. ¿Qué son los archivos gráficos? Indica los archivos gráficos más usados y explica brevemente sus características.

14. Responde las preguntas siguientes:

a) ¿Cuál es el formato de calidad en el que la imagen se forma mediante una parrilla de píxeles y no hay pérdida de calidad?

b) ¿Cuál es el formato utilizado para almacenar imágenes de una calidad excelente, que emplea cualquier profundidad de color de 1 bit a 32 bits y es ideal para editar imágenes de alta calidad que van a ser impresas?

15. ¿Qué formato de imagen utiliza GIMP de forma predeterminada? ¿Y Photoshop?

16. Investiga, ¿qué puede ocurrir si guardo una imagen con filtros y capas en formato JPG?

17. Explica con tus palabras qué entiendes por fotomontaje o montaje fotográfico. ¿Qué ventajas le atribuyes a su utilización en el mundo contemporáneo? Ten en cuenta su empleo en el diseño gráfico, el periodismo y la publicidad, de manera general.

18. Encuentra el concepto de cada una de las definiciones que aparecen a continuación.

Y	M	Ó	B	Q	O	W	H	S	B
H	C	Y	S	P	Ñ	C	M	Y	K
H	Y	M	Ó	Ñ	Y	Ó	Z	Í	Ú
G	Ú	F	Ú	P	Í	X	E	L	H
B	K	I	N	D	E	X	A	D	O
Ü	N	Z	Ó	N	R	Z	Á	T	W
G	Í	Ú	Á	M	D	R	G	B	E
L	T	K	Ü	T	Á	C	I	P	R
S	V	O	Í	Í	I	N	E	Á	Y
M	J	R	O	Y	S	N	X	T	R

Definición

- Son las unidades de color que componen la imagen.
- Modo de color que utiliza tres canales, donde cada canal se corresponde con un color primario: rojo, verde o azul.
- Modo de color que se forma. Cada color se forma por la combinación de cuatro canales.
- Modo donde cada color surge de los valores de tres parámetros: **Hue** (tono), **Saturation** (Saturación) y **Brightness** (Brillo).

19. Menciona las diferentes formas de obtener una imagen digital.
20. Si quisieras digitalizar un conjunto de fotos familiares que hace muchos años guardas en un álbum, ¿qué dispositivo utilizarías?
 - a) ¿Qué técnica utilizarías para que las imágenes queden almacenadas digitalmente como si las hubieras acabado de tomar?

21. Crea en tu ordenador un árbol genealógico de tu familia y coloca en el lugar de cada uno de los miembros de la familia una fotografía que lo identifique.
22. Indaga, ¿cuáles son los cinco editores gráficos gratuitos y de calidad que puedes utilizar en la edición de imágenes digitales?
a) Caracteriza dos de ellos y menciona las ventajas de cada uno.
23. ¿Qué formato de imagen utiliza GIMP de forma predeterminada? ¿Y Photoshop? ¿Qué puede ocurrir si se guarda una imagen con filtros y capas en formato JPG?
24. Investiga, ¿a qué familia pertenece Photoshop?, ¿quién lo distribuye y comercializa?, ¿cómo es su distribución?
25. Elabora un cuadro comparativo entre las aplicaciones Photoshop y GIMP donde aparezcan sus principales semejanzas, diferencias, ventajas y desventajas.
26. Investiga para qué se utiliza cada una de las herramientas de las ventanas de herramientas de GIMP y Photoshop.
27. Photoshop soporta archivos de tipo ICO, IFF, DCS, TGA, PCX, Scitex CT, Filmstrip, FlashPix y JPEG2000. Investiga para qué se utiliza cada uno de estos tipos de archivo.
28. ¿Qué herramientas de selección se pueden encontrar en GIMP? ¿Y de transformación?
29. Investiga qué quiere decir que una imagen está sobreexpuesta y subexpuesta. Explica con tus palabras cómo se puede corregir en GIMP.
30. Estudia en los materiales bibliográficos que se ponen a tu disposición y responde:
a) ¿Qué diferencia hay entre el tono, la luminosidad y la saturación de una imagen?

31. Explica qué son las capas y en qué casos, en general, es bueno usarlas.
32. Imagina que tienes que realizar un *colash* de imágenes, donde ilustres la belleza de los campos cubanos. ¿Qué editor de imágenes utilizarías? Justifica tu selección.
33. Abre una imagen cualquiera que poseas en tu PC, cámbiale el tamaño y la resolución. A continuación, guárdala en la carpeta que debes crear con tu nombre y grupo para el trabajo en la asignatura, atendiendo a los siguientes formatos:
- Imagen guardada en el formato nativo de GIMP "XCF".
 - Imagen guardada en el formato BMP.
 - Imagen guardada en el formato TIFF.
 - Imagen guardada en el formato GIF.
 - Imagen guardada en el formato JPG.
- a) Analiza y compara el tamaño de las imágenes. ¿Cómo son? ¿Qué formato ocupa más espacio? Explica con tus palabras a qué se debe esto.
34. Copia una de las imágenes que posees en tu PC hacia tu carpeta de trabajo de la asignatura y realiza las siguientes transformaciones con respecto a los modos de color. Conviértelas a:
- Mapa de bits. Menú Imagen » Modo » Indexado » Usar paleta en blanco/negro (1 bit). Y en las opciones de difuminado eliges **Ninguno**, después **Aceptar**. Guarda el resultado como "map_bits.bmp".
 - Escala de grises. Menú Imagen » Modo » Escala de grises. Guarda el resultado como "es_gris.bmp".
 - Color indexado. Menú Herramientas » Herramientas de color » Colorear. No modifiques ningún valor y haz clic en **Aceptar**. Después accede a menú Imagen » Modo » Indexado, y seleccionas **Generar paleta óptima**, y en las opciones de difuminado seleccionas **Ninguno**. Guarda el resultado como "col_index.bmp".
 - Color indexado. Menú Imagen » Modo » Indexado » Generar paleta óptima y No. máximo de colores 10. Guarda el resultado como "col_index_10.bmp".
 - Color RGB. Sin realizar ninguna modificación a la imagen, guarda el resultado como "col_rgb.bmp".

a) Compara las imágenes resultantes en cuanto al tamaño del archivo. Explica con tus palabras lo sucedido con los tamaños de los archivos, teniendo en cuenta los contenidos estudiados sobre modos y profundidad de color.

35. Copia una imagen de un paisaje cubano hacia tu carpeta de trabajo y realiza los ajustes necesarios al tono, color, brillo y contraste para obtener:

- Una imagen tomada en invierno.
- Una imagen tomada en otoño.

a) Guárdalas cada una en formato JPG y ponle como nombre la estación que representa.

36. Busca varias imágenes sobre nuestro líder histórico el Comandante en Jefe Fidel Castro Ruz y realiza un fotomontaje donde ilustres los grandes valores de este hombre, único en la historia de Cuba y el mundo. Primero debes retocar cada una de las imágenes, para luego realizar la infografía.

37. Realiza una excursión por los alrededores de la escuela y toma varias fotografías que ilustren la vegetación predominante, retócalas y únelas en un fotomontaje donde se evidencie la belleza de la vegetación cubana.

38. Elabora una infografía donde ilustres los rasgos de una vida saludable (ten en cuenta la alimentación saludable, la práctica de ejercicios físicos, la recreación y el rechazo a las adicciones).

39. Mira la imagen que aparece a continuación, realiza una infografía donde abordes las ventajas de la realización de actividades físicas regularmente.



40. Mira la siguiente imagen correspondiente al proceso de obsolescencia programada.



a) Investiga este fenómeno, caracterízalo y realiza una infografía donde expreses en qué consiste y cómo llega a afectar al medio ambiente.

1.2 Tratamiento del sonido digital

Todo sonido que se escucha es una superposición de “ondas” u oscilaciones de presión a distintas frecuencias. Dichas oscilaciones se transmiten por el aire en forma de variaciones de presión que pueden ser captadas por nuestros oídos o por dispositivos de captura de sonido como los micrófonos.

El sonido es una vibración que se propaga por el aire, pues las moléculas de aire transmiten la vibración hasta nuestros oídos. Es la interpretación que hace nuestro cerebro de las variaciones de presión que genera un objeto vibrante en determinado medio (habitualmente, el aire).



Definición

Esta perturbación que se transmite por el aire se llama **onda sonora** y la forma que esta adopta se conoce como **forma de onda**.

En los medios analógicos, las ondas u oscilaciones que constituyen el sonido se registran sobre un soporte físico como protuberancias o surcos (como en un disco de vinilo), como variaciones del campo magnético (como en una cinta de casete) o como variaciones de luz (como en el celuloide de las películas). Esas oscilaciones serán captadas posteriormente por un dispositivo lector (la cabeza lectora de un magnetófono, la aguja de un tocadiscos de vinilo, o el fotodetector de un proyector de cine) y reforzadas por un amplificador para su emisión sonora a través de altavoces o auriculares, que son los dispositivos finales que vuelven a generar las oscilaciones de presión de aire que nuestro oído es capaz de captar.

Estos son los mecanismos de grabación de sonido que se han venido utilizando durante años, hasta la aparición de la tecnología digital.

Aunque los medios de grabación analógica de sonido consiguen una excelente calidad, las señales analógicas son más difíciles de procesar que las digitales, y los dispositivos necesarios para manejarlas manteniendo esa calidad resultan mucho más caros. La digitalización del sonido aporta grandes ventajas tanto técnicas como funcionales. Por otra parte, los avances tecnológicos de los últimos años han abaratado extraordinariamente el costo de esta alternativa, hasta el punto de que hoy podemos contar en nuestro ordenador con unas capacidades de grabación y tratamiento de sonido que hace unos años solo estaban al alcance de carísimos estudios

de sonido profesionales. El sonido tratado digitalmente puede copiarse, guardarse y reproducirse infinitamente sin perder calidad; además puede procesarse de manera mucho más flexible y potente que mediante procesos analógicos.

El sonido en su origen es una **señal analógica** que debe ser convertida en **señal digital**, para que pueda ser admitida por un sistema informático, que solo entiende de señales eléctricas.



Definición

El **sonido digital** o **audio** es la representación de una secuencia de sonidos convertida al formato binario para poder ser almacenada en un ordenador.

Un sistema completo de audio digital comienza habitualmente con un transceptor (micrófono) que convierte la onda de presión que representa el sonido en una señal eléctrica analógica. Esta señal analógica atraviesa un sistema que procesa dicha señal, en el cual se pueden realizar limitaciones en frecuencia, ecualización, amplificación y otros procesos como el de compensación.

La **ecualización** se realiza para contrarrestar la particular respuesta en frecuencia del transceptor utilizado, de forma que la señal analógica se asemeje mucho más a la señal de audio originaria. Tras el procesado analógico la señal se muestrea, se cuantifica y se codifica.

Digitalizar el sonido consiste en transformar la información de las ondas en secuencias de números binarios, que son los que un ordenador sabe interpretar. El proceso de digitalización se basa en un concepto denominado “muestreo”, que consiste a grandes rasgos en lo siguiente: en lugar de almacenar los infinitos puntos de información que definen una onda (como hacen los soportes analógicos al registrar físicamente la onda continua del sonido sobre un determinado soporte), se toman “medidas” (o “muestras”) periódicas de esa onda, y los valores en esos puntos se convierten en números digitales.



Definición

La palabra **muestreo** es el equivalente del término inglés *sampling*, y se utiliza para indicar la acción de tomar muestras a intervalos de tiempo regulares.

Para digitalizar un sonido es necesario muestrearlo, pero ¿con qué frecuencia?



Definición

La **frecuencia de muestreo** es un número que indica la cantidad de muestras que se toman en determinado intervalo de tiempo; por ejemplo, en un segundo.

Existe una ley matemática conocida como teorema de Nyquist que establece que con una frecuencia de muestreo F es posible reproducir fielmente cualquier señal que contenga frecuencias de sonido de hasta $\frac{F}{2}$:

Ejemplo

Si muestreamos a 44 KHz, podremos reproducir sonido con frecuencias de hasta 22 KHz, más que suficiente para las capacidades auditivas de un ser humano medio. Esta es, por cierto, la frecuencia de muestreo utilizada en los CD comerciales.

Matemáticamente, se ha demostrado que si la velocidad a la que se toman las muestras (o “frecuencia de muestreo”) es lo bastante elevada, es posible reconstruir con exactitud toda la señal original a partir de esos puntos muestreados (es decir, que se puede deducir el valor de los infinitos puntos situados entre muestra y muestra conociendo solo los valores en los puntos muestreados).



Definición

La **resolución o profundidad del sonido** es un número que indica cuántos bits (dígitos binarios, ceros y unos) se utilizan para representar cada muestra.

Tanto la frecuencia como la resolución están directamente relacionadas con la calidad del sonido digital almacenado. Mientras mayores sean estos indicadores, más parecida será la calidad del sonido digitalizado con respecto al real.

El estándar definido cuando se crearon los discos compactos de audio especifica que el sonido digital almacenado en ellos debe poseer una frecuencia de 44,1 KHz (KiloHertz) y 16 bits estéreo. Esto significa que se

deben tomar unas 44 100 muestras por segundo, cada una se representará con 16 bits, y en dos canales independientes (sonido estéreo).



Saber más

Si se quiere saber cuántos ceros y unos se necesitan para almacenar tres minutos de sonido digital con calidad CD (44,1 kHz y 16 bits estéreo), se debe realizar el siguiente cálculo:

- $3 \text{ min} \times 60 \text{ s} = 180 \text{ s}$
- $180 \text{ s} \times 44\,100 \text{ muestras por segundo} = 7\,938\,000 \text{ muestras}$
- $7\,938\,000 \text{ muestras} \times 16 \text{ bits} = 127\,008\,000 \text{ bits}$
- $127\,008\,000 \text{ bits} \times 2 \text{ canales} = 254\,016\,000 \text{ bits}$

Entonces para almacenar una canción de 3 min con calidad de CD se necesitarían 254 016 000 ceros y unos o lo que es igual, 31 752 000 bytes (unos 30 Mb).

Todo lo analizado anteriormente significa, en otras palabras, que nos basta una cantidad finita de información para representar un número infinito de puntos. El proceso de digitalización se conoce como Conversión Analógico-Digital (A/D), mientras que el proceso inverso es la Conversión Digital-Analógico, que consiste en transformar la información numérica discreta (“a trozos”) en ondas sonoras continuas. En un ordenador, estos procesos los realiza la tarjeta de sonido.



Algo de historia

En 1982 aparece el disco compacto o CD, que inaugura la era del sonido digital para el gran público. Hasta entonces, los soportes de grabación a través de los cuales se distribuía la música eran analógicos, como las cintas de casete o los discos de vinilo.

1.2.1 Características del sonido digital

El espectro de un sonido se caracteriza por su rango de frecuencias. Esta se mide en Hertzios (Hz). El oído humano capta solo aquellos sonidos comprendidos en el rango de frecuencias entre 20 Hz y 20 000 Hz.

Para describir un sonido “musical” se utilizan tres términos: altura, timbre e intensidad. La altura está directamente relacionada con la frecuencia

de oscilación, aunque algunos sonidos, como los percusivos, no tienen una altura definida. Lo que hace que un sonido posea o no una altura clara es la periodicidad de la señal. Es decir, que para que se llegue a percibir su altura es necesario que la frecuencia de oscilación (que se mide en Hz —cantidad de ciclos de la señal por segundo—) no varíe dentro de un determinado lapso mínimo de tiempo. Mientras más alta sea la frecuencia de la señal, más agudo será el sonido, y viceversa.



Definición

La **frecuencia de audio** (altura) es el número de vibraciones por segundo que da origen al sonido analógico.

Un audio digital es una secuencia de ceros y unos que se obtiene del muestreo de la señal analógica.



Definición

La **amplitud** (intensidad, volumen) es la cantidad de fuerza o energía de un sonido.

La **intensidad o amplitud** depende de la diferencia entre las presiones máxima y mínima que la onda puede alcanzar. Se puede asociar al volumen del sonido, pero la percepción de la intensidad es un fenómeno auditivo muy complejo, por lo cual se indica que mientras mayor sea la amplitud de la onda, mayor será el volumen y viceversa. La medida de la amplitud es logarítmica y se hace en decibelios dB. Subir 3 dB significa que se duplica la potencia de un sonido.



Definición

El **timbre** es lo que nos ayuda a caracterizar y distinguir diferentes tipos de instrumentos, o a reconocer a las personas por su voz (figura 1.92 a, b).

Por ejemplo, dos señales pueden tener la misma frecuencia (altura) pero diferente forma: timbre (figura 1.93 a, b).

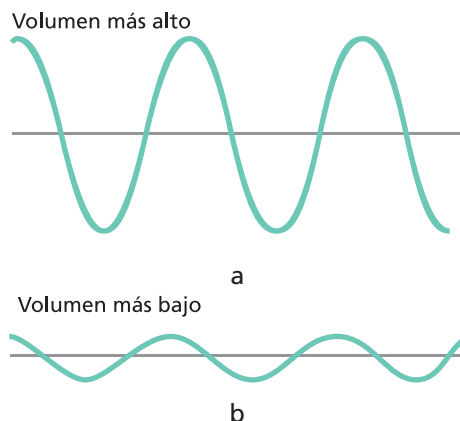


Fig. 1.92

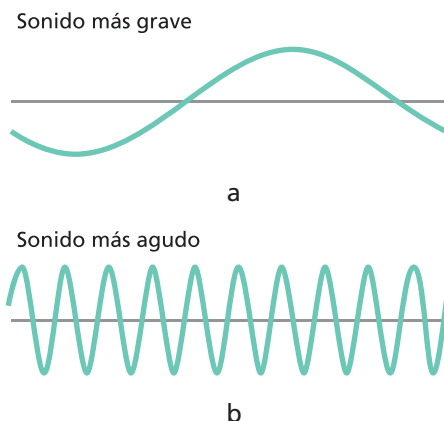


Fig. 1.93

Esto se debe a que las ondas de los sonidos naturales son complejas porque vibran con varias frecuencias simultáneas. La frecuencia de vibración más grave es la que determina normalmente el período y la altura, y se denomina **frecuencia base**.

A continuación, se estudiarán otras características del sonido digital relacionadas con la intensidad o amplitud de este.



Definición

Rango dinámico, es la diferencia entre el valor mínimo y máximo que puede tener un sonido.

Ancho de banda, es la diferencia de las frecuencias en la que se produce una caída de la amplitud del sonido determinada (se suele usar 3 decibelio).

El **decibelio** es la unidad de medida del volumen o intensidad de un sonido. El silencio o ausencia de sonido se cuantifica como 0 dB y el umbral del dolor para el oído humano se sitúa en torno a los 130 dB-140 dB.

CBR/VBR (**Constant/Variable Bitrate**). En estas siglas CBR indica que el audio ha sido codificado manteniendo el bitrate constante a lo largo del clip de audio, mientras que VBR varía entre un rango máximo y mínimo en función de la tasa de transferencia.



Definición

Ruido, son sonidos aleatorios que proceden de diversas fuentes y distorsionan o enmascaran el sonido fundamental. Se mide en dB.



Definición

Resolución (bit resolution), es el número de bits utilizados para almacenar cada muestra de la señal analógica.

Una resolución de 8 bits proporciona 256 (2^8) niveles de amplitud, mientras que una resolución de 16 bits alcanza 65 536 (2^{16}). Un audio digital tendrá más calidad cuanto mayor sea su resolución. Por ejemplo, el audio de calidad del CD suele ser un sonido de 44 100 Hz – 16 bits – estéreo.



Definición

Velocidad de transmisión (bitrate), el bitrate define la cantidad de espacio físico (en bits) que ocupa un segundo de duración de ese audio.

Por ejemplo: 3 min de audio MP3 a 128 Kbit/seg, ocupa 2,81 Mb de espacio físico ($3 \text{ min} \times 60 \text{ seg/min} \times 128 \text{ Kbit/seg} = 23\,040 \text{ Kbits} \rightarrow 23\,040 \text{ Kbits} \times 1\,024 \text{ bits/Kbit} : 8 \text{ bits/bytes} : 1\,024 \text{ bytes/Kbytes} : 1\,024 \text{ Kbytes/Mbytes} = 2,81 \text{ Mbytes o Mb}$).

En los audios en formato MP3 se suele trabajar con bitrates de 128 Kbps (kilobits por segundo). El audio tendrá más calidad cuanto mayor sea su bitrate, y el archivo que lo contiene tendrá mayor peso. Esta magnitud se utiliza sobre todo en el formato MP3 de audio, más destinado a la descarga por Internet.



Definición

Tasa de muestreo (sample rate) define cada cuánto tiempo se tomará el valor de la señal analógica para generar el audio digital. Esta tasa se mide en Hertzios (Hz).

Por ejemplo, 44 100 Hz nos indica que en un segundo se tomaron 44 100 muestras de la señal analógica de audio para crear el audio digital correspondiente.

Un audio tendrá más calidad cuanto mayor sea su tasa de muestreo. Algunas frecuencias estándares son 44 100 Hz, 22 050 Hz y 11 025 Hz.

1.2.1.1. Formatos de audio

Al igual que con las imágenes, la información digital del sonido puede almacenarse comprimida o sin comprimir. La información del sonido sin comprimir recoge directamente los valores muestreados.

Para una frecuencia de muestreo de 44 KHz (la utilizada en los CD de música), se tendrán 44 000 valores muestreados por segundo. Cada uno de esos valores suele representarse con un número binario de 16 bits (2 bytes). Si la grabación es en estéreo (con dos canales, izquierdo y derecho), la cantidad de información se duplica. Basta con hacer números para darse cuenta de la gran cantidad de espacio que ocupa el sonido sin comprimir.

Sin embargo, es posible reducir considerablemente esa cantidad de información aplicando técnicas de compresión de sonido, que se apoyan en las características psicoacústicas del sistema auditivo humano para aprovechar redundancias en el sonido o eliminar frecuencias que nuestro oído o nuestro cerebro no pueden captar. Este proceso suele suponer una cierta pérdida de calidad, pero para la mayoría de las personas resulta aceptable. Un ejemplo de ello es el popular formato MP3.

La información del sonido se guarda en el ordenador en archivos con un determinado formato, el cual define las características y el tipo de codificación del archivo. Por tanto, el formato en el que está almacenado el archivo depende del códec utilizado para su codificación. Los más extendidos son el formato WAV (o CDA audio en calidad CD) utilizado en Windows, el formato MP3 (*Mpeg layer 3*), AAC (*Apple Ipod*), WMA (*Windows Media Audio*), AU (*Audio for Unix*), AC (*Sistemas Dolby* de alta calidad), MIDI (en inglés: *instrumento digital de interfaz digital*), OGG (*Vorbis*, similar al .mp3 pero libre y el ATRAC (para minidiscos, PDAs y teléfonos móviles). Dentro del formato WAV, existen distintas variantes o subformatos, que se diferencian en el método de codificación utilizado (códec).

Existen códec sin compresión (y por tanto sin pérdida de calidad) como PCM (*Pulse Coded Modulation*, de calidad equivalente a un CD de música), y códec con compresión como .mp3. Pero te estarás preguntando *¿qué es un códec?*



Definición

Códec. Acrónimo de “codificación/decodificación”. Un códec es un algoritmo especial que reduce el número de bytes que ocupa un archivo de audio. Los archivos codificados con un códec específico requieren el mismo códec para ser decodificados y reproducidos. El códec más utilizado en audio es el MP3.

Es importante diferenciar entre el formato del archivo de sonido (por ejemplo, WAV), que define la organización de la información dentro del archivo, y el códec utilizado, que determina el método de compresión o digitalización empleado dentro de ese archivo. Por ejemplo, un archivo WAV puede utilizar distintos códec. Uno de los más comunes en los WAV es el PCM, sin compresión, pero también pueden utilizarse otros, y el códec en cuestión debe estar instalado en el sistema operativo para que el archivo funcione.

El formato de archivo .mp3 utilizado habitualmente para la música en Internet es un tipo de formato con un códec fijo, el .mp3; aunque con distintas variantes según el nivel de compresión. Ese mismo códec .mp3 puede utilizarse también dentro de un archivo WAV.



El **formato MP3 (Mpeg layer 3)** fue creado por el Instituto Fraunhofer junto a los formatos MPEG, que dieron lugar al DVD. Por su extraordinario grado de compresión y alta calidad es el formato más extendido, ya que consigue compresiones de 11 a 1 prácticamente sin pérdidas de calidad. Comprime por eliminación de escalas sonidos que el oído humano no puede oír, lo que hace que se reduzca la cantidad de bytes necesarios en el archivo final.

Es ideal para publicar audios en la Web y se puede escuchar desde la mayoría de los reproductores.



Saber más

Los bitrates adecuados para el formato MP3 son:

- Grabar conversación o discurso: 8 Kbits/s
- Grabar Radio AM: 32 Kbits/s
- Grabar Radio FM: 96 Kbits/s
- Extraer pistas de CD en calidad media: 128 Kbits/s
- Extraer pistas de CD en calidad alta: 192 Kbits/s
- Extraer pistas de CD con la máxima calidad posible para el formato MP3: 320 Kbits/s



El **formato WMA** fue diseñado por Microsoft para su reproductor de Windows Media. Muy similar al MP3.



El **formato WAV** (*WaveForm Audio File*) es un formato de archivo que desarrolló originalmente Microsoft e IBM, para guardar audio. Los archivos tienen extensión ***.wav** y son ideales para guardar audios originales que se pueden comprimir y guardar en distintos tamaños de muestreo para publicar en la Web. Es un formato de excelente calidad de audio, aunque produce archivos de un peso enorme. Los archivos WAV se pueden guardar con distintos tipos de compresión. Las más utilizadas son la compresión PCM y la ADPCM. No obstante, incluso definiendo un sistema de compresión, con un audio de cierta duración, se genera un archivo excesivamente pesado. Su logaritmo de codificación comprime sin pérdidas. Es el formato en los que vienen grabados los CD, con una frecuencia de muestreo de 44,1 KHz y codificados a una resolución de 16 bits.



Saber más

Una canción extraída de un CD (16 bytes, 44 100 Hz y estéreo) puede ocupar entre 20 Mb y 30 Mb.

El formato WAV se suele utilizar para fragmentos muy cortos (no superiores a 3 s-4 s), normalmente en calidad mono y con una compresión Microsoft ADPCM 4 bits.



El **formato ACC** es una codificación de audio reconocida por ISO en el patrón MPG-2. En teoría, almacena más que el MP3 en menos espacio. Este es el formato de audio que utiliza Apple para los archivos de audio que reproduce el iTunes y el iPod.



El **formato OGG VORBIS** ha sido desarrollado por la Fundación Xiph.org. La principal diferencia de este formato con el resto es que es gratuito, abierto y no está patentado. Es el formato más reciente y surgió como alternativa libre y de código abierto (a diferencia del formato MP3). Muestra un grado de compresión similar al MP3 pero según los expertos en música la calidad de

reproducción es ligeramente superior. Su principal atractivo es la importante reducción que hace a un archivo de audio sin quitarle calidad y se distingue por su versatilidad para reproducirse en prácticamente cualquier dispositivo y por ocupar muy poco espacio. Pero no todos los reproductores multimedia son capaces de leer por defecto este formato, por tanto, en algunos casos es necesario instalar los códec o filtros oportunos. Este formato puede contener audio y video.

A continuación, se analizará especialmente el formato MIDI. Aunque no es un formato de audio propiamente dicho, se estudiarán sus características.



El **formato MIDI** (*Musical Instrument Digital Interface* = *Interface Digital para Instrumentos Digitales*) en realidad no resulta de un proceso de digitalización de un sonido analógico. Es considerado el formato estándar para industrias de la música electrónica. También es muy útil para trabajar

con dispositivos como sintetizadores musicales o tarjetas de sonido. Un archivo de extensión *.mid almacena secuencias de dispositivos MIDI (sintetizadores) donde se recoge qué instrumento interviene, en qué forma lo hace y cuándo.

Los archivos MIDI permiten audios de cierta duración con un reducido peso. Esto es debido a que no guardan el sonido sino la información o partitura necesaria para que el ordenador la componga y reproduzca a través de la tarjeta de sonido. Por el tamaño resultante que ofrece su compresión, este formato es muy usado para reproductores que necesitan combinar archivos de audio y video como los karaokes. Este formato es interpretado por los principales reproductores del mercado: Windows Media Player, QuickTime, etc. Los archivos MIDI se pueden editar y manipular mediante programas especiales y distintos a los empleados para editar formatos WAV, MP3, etc. El trabajo de estos programas suele conllevar ciertos conocimientos musicales.

Observación

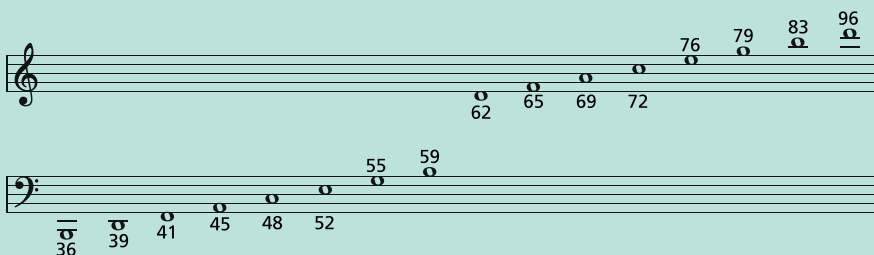
El formato MIDI no permite la riqueza de matices sonoros que otros formatos ni la grabación a partir de eventos sonoros analógicos.



Saber más

Las notas musicales se representan en MIDI mediante un número entero positivo comprendido entre 0 y 127 (requiere de 7 bits). Con este rango cubre 11 octavas. Las correspondencias son las siguientes:

Octava	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
	Do		Re		Mi	Fa		Sol		La		Si
0	0	1	2	3	4	5	6	7	8	9	10	11
1	12	13	14	15	16	17	18	19	20	21	22	23
2	24	25	26	27	28	29	30	31	32	33	34	35
3	36	37	38	39	40	41	42	43	44	45	46	47
4	48	49	50	51	52	53	54	55	56	57	58	59
5	60	61	62	63	64	65	66	67	68	69	70	71
6	72	73	74	75	76	77	78	79	80	81	82	83
7	84	85	86	87	88	89	90	91	92	93	94	95
8	96	97	98	99	100	101	102	103	104	105	106	107
9	108	109	110	111	112	113	114	115	116	117	118	119
10	120	121	122	123	124	125	126	127				



1.2.2 Componentes del hardware, necesarios para manejar el sonido digital

Es esencial aclarar que aquí no se hará un análisis de los distintos componentes que intervienen en el procesamiento del sonido digital; solamente se van a enumerar los más importantes. Además, se dejará bien claro que se hace referencia a un procesamiento de sonido relativamente simple con la utilización de una computadora personal, no a lo que necesita un trabajo profesional.

Para lograr escuchar el sonido digital, en primer lugar, se necesita una PC con una tarjeta de sonido, actualmente todas las PC ya traen tarjeta de sonido **on board** (en la tarjeta madre o **Motherboard**). Esta tarjeta será la encargada de convertir la información digital en una señal eléctrica que luego será enviada a los altavoces (o bocinas) para ser convertida finalmente en vibraciones (el sonido que escuchamos). Eventualmente, la tarjeta de sonido también es la encargada de realizar el proceso contrario, o sea, la digitalización del sonido explicada anteriormente. Luego se necesitaría de altavoces o auriculares y, en algunas ocasiones, de un amplificador entre la placa de sonido y los altavoces, para lograr una mayor potencia.

Ahora bien, para digitalizar nuestra voz o el sonido del ambiente, podemos utilizar un micrófono. Debes tener presente que muchas placas de sonido admiten conectarles como entrada, la salida de algún otro equipo de audio y, en general, el mismo lector de CD o DVD puede utilizarse para leer el sonido de un disco compacto.

Como verás existen varios tipos de captura: la que se realiza a partir de un micrófono, que se efectúa por una entrada analógica y la de entrada digital. A continuación, se expondrán algunos aspectos que se deben tener en cuenta en cada uno de estos tipos de captura:

1. Captura a través de micrófono: es necesario tener en cuenta el tipo de micrófono que se va a utilizar:
 - Omnidireccional o direccional
 - De mayor o menor sensibilidad
2. Por entrada analógica: es necesario saber:
 - Fuente analógica:
 - Relación S/R baja → Ruido
 - Medios deteriorados

- Fuente digital:
 - Relación S/R alta
 - Pérdidas e inducción de ruido en la conversión
- 3. Por entrada digital: es necesario saber:
 - La máxima calidad
 - No debe haber interferencias, ruido ni conversión



Saber más

Para realizar capturas analógicas, te ofrecemos los siguientes consejos:

1. En cuanto a los **micrófonos** debes:

- Usar tipos adecuados para cada situación.
- Usar filtros (filtros antipop, pantallas, etcétera).
- Tener en cuenta la cercanía a la fuente sonora.
- Ambientes con poco ruido de fondo. Mejora S/N.
- Evitar saturación de niveles.
- Alejar del ordenador – cable de calidad – pantalla.

2. En cuanto a las **fuentes** debes:

- Alejar la fuente del ordenador > menos interferencias > cables de calidad.
- Ajustar niveles de entrada al máximo sin saturar.
- Subir mejor el nivel de salida que el de entrada.

Otra de las tareas más frecuentes en la elaboración de un material multimedia con sonido digital es la extracción de la información de un CD para su conversión en sonido digital en formato WAV o .mp3. Esto nos permite, por ejemplo, poder oír la música de nuestros CD en uno de los reproductores de mp3, tan populares hoy en día, o incluso en el teléfono móvil, ya que la mayoría de los modelos modernos son capaces de reproducir archivos de sonido en este formato.

En la jerga informática, extraer canciones de un CD a un archivo se conoce como “ripear” el CD. Para realizar esta operación existen muchos programas. Para plataformas Windows, hemos elegido Express Rip por su sencillez y porque es gratuito. Una vez instalado, al ejecutar el programa, el operador encontrará el Entorno de trabajo (figura 1.94).

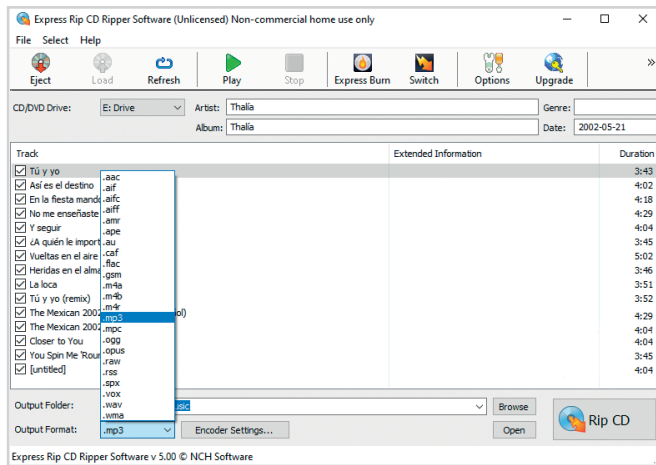


Fig. 1.94

Al introducir el CD en la unidad de lectura, el programa identifica automáticamente la fuente y aparece en la parte central del Entorno de trabajo un listado con todas las pistas del disco. Algunos CD aportan información sobre el nombre de la pista. Si ninguno de estos métodos permite determinar el título de las pistas, aparecen todas con el nombre **Track**, seguido de un número correlativo. Este nombre se puede cambiar haciendo clic sobre la pista con el botón derecho del ratón. También hay dos barras para escribir, si se desea, el nombre del álbum y el autor.

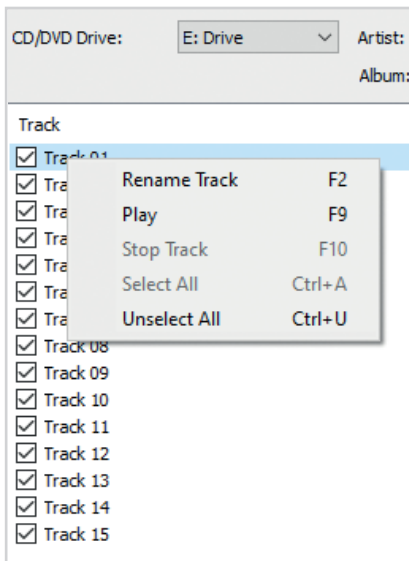


Fig. 1.95

Como se puede ver, a la izquierda del nombre de cada pista hay una casilla de verificación, que por defecto está activada, pero que se puede desactivar si se desea omitir de la extracción una determinada pista (figura 1.95).

Antes de proceder a la extracción de las pistas seleccionadas hay que indicar al programa la carpeta de destino (**Output Folder**) de los archivos extraídos, así como el formato con el que se guardarán (**Output Format**). Haciendo clic en el botón **Browse**, aparecerá una ventana que

permite recorrer las distintas carpetas existentes en el disco y seleccionar una. El botón **Open** abre directamente el Explorador de Windows.

Una vez elegida la carpeta de destino, y salvo que se quieran modificar otras variables más avanzadas, no hay más que pulsar el botón **Rip CD** para comenzar el proceso de extracción (figura 1.96).

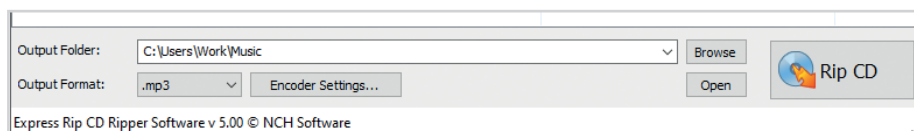


Fig. 1.96



Cómo se hace

Para extraer pistas de audio de un CD:

1. Se carga el CD.
2. Se cambia el nombre de las pistas, si se desea.
3. Se seleccionan las pistas que se quieren extraer.
4. Se elige la carpeta de salida y el formato de los archivos de sonido que se generarán.
5. Luego comienza la extracción.

1.2.3 Procesamiento digital del sonido

Bueno, una vez que se tiene guardado en el disco de nuestra PC un archivo de sonido, podemos “retocarlo” y aplicarle diferentes efectos, generalmente llamados filtros. Esto es posible hacerlo desde operaciones sencillas, como quitar una porción o repetir otra, modificar el volumen, etc., hasta otras más complejas como destacar ciertos matices, aplacar otros, etcétera.

Como debes imaginarte, con los constantes adelantos tecnológicos existe una amplia gama de software para procesamiento de archivos de sonido, desde la sencillísima “Grabadora de sonido” de Windows, pasando por el CoolEdit o el Audacity, hasta programas más complejos como el SoundForge o el ahora llamado Adobe Audition.

La edición del audio contempla una serie de operaciones técnicas que la mayoría del **software** usa y que pueden ser transparentes para el usuario, por lo que a continuación se explican algunos términos usados:

- **Amplificación de sonido:** la operación de amplificación aumenta la amplitud de la onda, el efecto en el espectador es el aumento del volumen (figura 1.97).
- **Inversión del sonido:** las crestas de las ondas se transforman en valles y viceversa. Si se invierten dos canales en una grabación estéreo no se percibe efecto alguno, pero si se invierte un solo canal, se puede percibir un cambio interesante, provocando cierto tipo de efecto especial (figura 1.98).

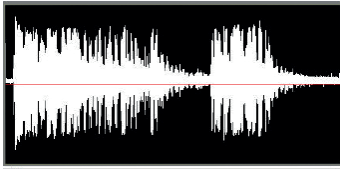


Fig. 1.97

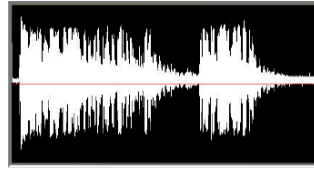


Fig. 1.98

- **Retraso (*Delay*):** se obtiene al retrasar el sonido una determinada cantidad de tiempo. Si se trabaja con una grabación monoaural se podría separar esta en dos canales idénticos y después desfazar un canal haciendo un retraso de milisegundos, esto crearía un efecto espacial, de manera que se percibiría que la fuente de sonido está ubicada a un lado de nosotros, ya que el cerebro estaría percibiendo que en uno de los oídos llega el sonido ligeramente atrasado con respecto a otro. Para percibir este efecto es necesario el uso de audífonos (figura 1.99).
- **Eco y cámara de eco:** este efecto surge cuando un sonido rebota en paredes dentro de una cavidad, pudiendo rebotar el sonido varias veces. Cada vez que rebota el sonido puede atenuarse de acuerdo con el grado de absorción de las vibraciones sonoras en las paredes. Los factores más importantes que determinan el efecto son el tamaño del cuarto, el número de ecos, factores de amortiguamiento de las paredes, posición del micrófono, posición de la fuente sonora, entre otros. Con este efecto también es posible transformar un sonido mono en estéreo al separar en canales y hacer un tratamiento independiente a los canales (figura 1.100).

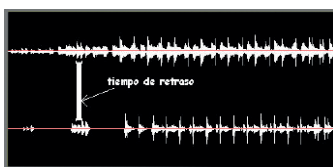


Fig. 1.99

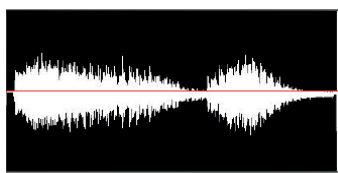


Fig. 1.100

- **Envolvente:** es posible transformar la amplitud de toda la onda en un intervalo de tiempo, multiplicando esta por el valor de una función que tenga una forma definida. Entonces el volumen del sonido resultante tendrá las mismas variaciones que la función que sirvió para multiplicar el sonido inicial, visualmente la gráfica de la onda de sonido resultante se observa como si estuviera envuelta en la forma que tenía la función que aplicamos como factor (figura 1.101).
- **Reversa:** tiene el efecto de desplegar la señal interpretándola de atrás para adelante (figura 1.102).

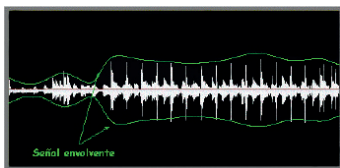


Fig. 1.101

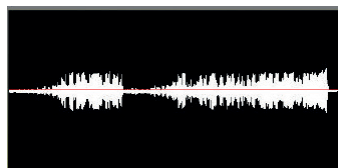


Fig. 1.102

La edición de audio es la manipulación de sonidos digitalizados; cortando y pegando partes, adicionando efectos especiales, mezclando sonidos. Para llevar a cabo esta tarea, se puede disponer tanto de un **software** comercial como de uno de dominio público o **freeware**. A continuación, se expondrá un representante de cada uno de estos tipos de **software**.



El **Adobe Audition** (anteriormente Cool Edit Pro) es una aplicación en forma de estudio de sonido destinada para la edición de audio digital de Adobe Systems Incorporated, que permite tanto un entorno de edición mezclado de ondas multipista no destructivo como uno destructivo, por lo que se le ha llamado la “navaja suiza” del audio digital por su versatilidad. No es DAW, sino un editor de sonido.



El **Audacity** es un editor de audio gratuito. Permite grabar y reproducir sonidos, importar y exportar archivos WAV, AIFF y MP3, entre otros; editar usando **cortar**, **copiar** y **pegar** (con ilimitados **deshacer**); mezclar pistas o aplicar efectos a las grabaciones. Audacity es desarrollado por un grupo de voluntarios bajo el modelo de código fuente abierto (**open-source**), por lo que el código fuente del programa está disponible bajo la licencia GNU General Public License, la cual esencialmente permite a cualquiera modificar el código fuente mientras publique los cambios.

Puedes usar Audacity para:

- Grabar sonidos en vivo.
- Convertir grabaciones a sonido digital o CD.
- Editar Ogg Vorbis, MP3 y WAV.
- Cortar, pegar, empalmar y mezclar sonidos.
- Cambiar la velocidad o el tono de una grabación.
- Aplicar efectos.

Con el sonido digital se pueden realizar disimiles acciones, una de las más importantes es la **optimización de archivos de audio** de acuerdo al fin o la utilización que el usuario desee darle, esta optimización incluye la reducción o el cambio de alguno de sus componentes o parámetros, como son:

- Tasa de muestreo: definir valores inferiores: 44 100 Hz, 22 050 Hz, 11 025 Hz, etcétera.
- Resolución: establecer resoluciones más pequeñas: 32 bits, 16 bits, 8 bits, 4 bits, etcétera.
- Duración: en ocasiones se puede utilizar un fragmento más corto que reproducido en bucle cubre el tiempo suficiente de acompañamiento musical. A estos se les llama **loops**.
- Calidad estéreo/mono: la reducción a calidad "mono" reduce considerablemente el peso del archivo. Por otro lado la calidad de reproducción "mono" para la mayoría de los audios y de público es apenas perceptible.
- Formato: es preferible utilizar el formato MP3 en lugar del WAV por su potente factor de compresión y su aceptable calidad de audio.
- Factor de compresión: el formato WAV admite distintos factores de compresión: PCM y ADPCM.

A continuación, se describirán los procedimientos necesarios para realizar estas tareas sobre un audio original, utilizando el editor Audacity. Es importante que te familiarices con su entorno de trabajo. Este cuenta con los siguientes elementos (figura 1.103):

1. Barra de menú.
2. Barra de herramientas.
3. Barra de herramientas de reproducción.
4. Barra métrica de grabación.
5. Barra métrica de reproducción.
6. Barra mezcladora.
7. Barra de edición.
8. Barra de copia o transcripción.
9. Barra de dispositivo.
10. Línea de tiempo.
11. Panel de control de secuencia.
12. Secuencia de audio.
13. Barra de selección.
14. Barra de estado.

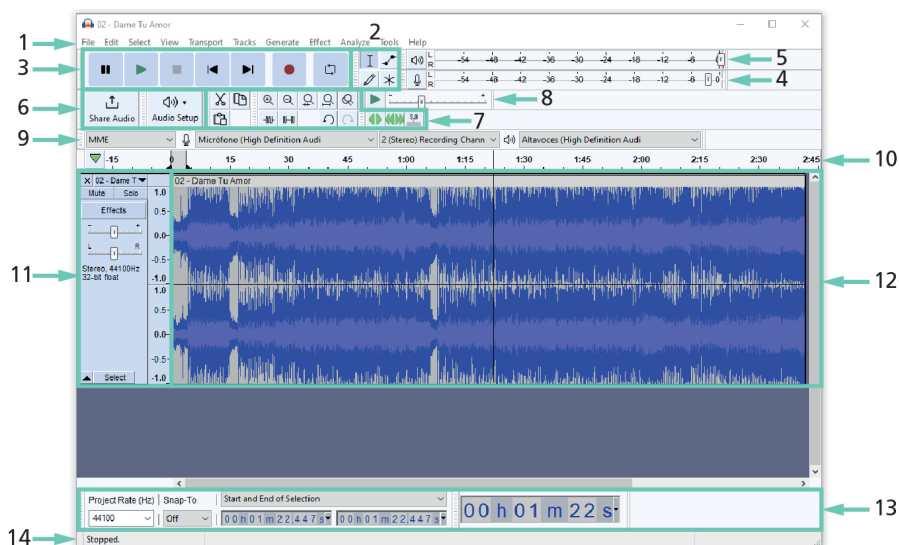


Fig. 1.103

Ahora podrás ver las opciones de los diferentes menús de la aplicación. Se comenzará con el menú **Archivo**, donde se pueden crear nuevos archivos, abrir existentes, guardar de diferentes formas, cerrar las pistas de audio, importar, exportar y aplicar, y editar secuencias de comandos (figura 1.104).

Ya en el menú **Editar** se encuentran las acostumbradas opciones de copiar, cortar, pegar, además de las de borrar, deshacer, rehacer, recortar límites, seleccionar, mover cursor, área de reproducción y preferencias (figura 1.105).

Archivo	Editar	Seleccionar	Ver	Repr
Nuevo			Ctrl+N	
Abrir...			Ctrl+O	
Archivos recientes				>
Cerrar			Ctrl+W	
Guardar proyecto				>
Exportar				>
Importar				>
Configurar página...				
Imprimir...				
Salir			Ctrl+Q	

Fig. 1.104

Editar	Seleccionar	Ver	Reproducción
Deshacer	Importar		Ctrl+Z
Rehacer			Ctrl+Y
Cortar			Ctrl+X
Borrar			Ctrl+K
Copiar			Ctrl+C
Pegar			Ctrl+V
Duplicar			Ctrl+D
Eliminar especial			>
Bloques de Audio			>
Etiquetas			>
Audio etiquetado			>
Metadatos			
Renombrar bloque...			Ctrl+F2
Preferencias			Ctrl+P

Fig. 1.105

En el menú **Ver**, se dispone de las opciones de visualización, ampliar, reducir, etc.; así como las de desplazarse al inicio o final de la selección, contraer y expandir las pistas, mostrar recorte, karaoke, panel mezclador y la opción de las barras de herramientas que se van a mostrar (figura 1.106).

En el menú **Control**, podrás determinar el inicio o fin de la reproducción, pausarla, grabar, grabación programada, añadir grabación, sobregrabar, reproducción a través del **software**, grabación por activación de sonido, nivel de activación y buscar de nuevo dispositivo de audio (figura 1.107).

El menú **Pista** te permitirá añadir nuevas pistas, mezclarlas, remuestrear, eliminar, silenciar, alinear, mover, enlazar pistas; editar etiquetas y ordenar las pistas (figura 1.108).

En el menú **Generar** podrás agregar ruidos, silenciar, seleccionar tonos, entre otras opciones (figura 1.109).

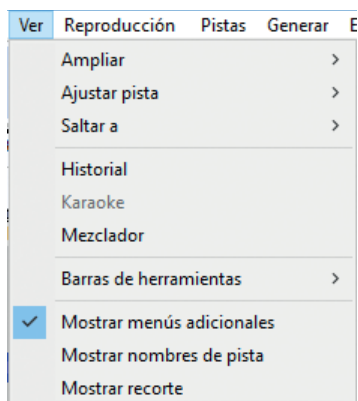


Fig. 1.106

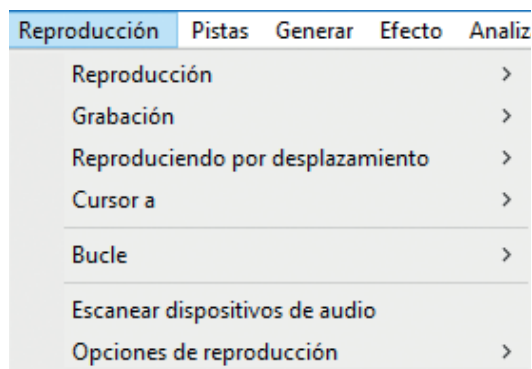


Fig. 1.107

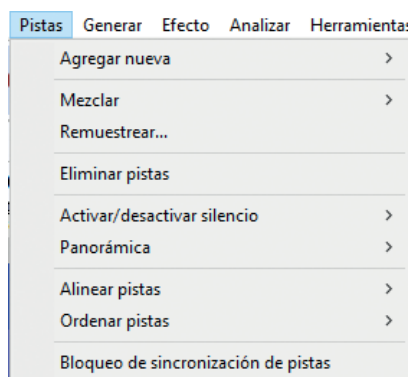


Fig. 1.108

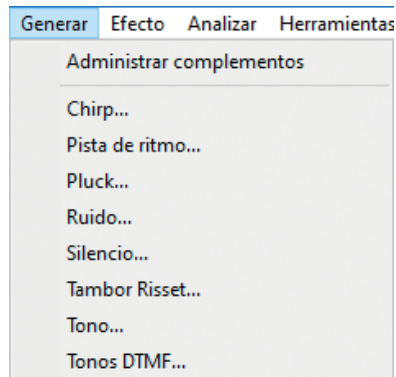


Fig. 1.109

En el menú **Efecto** encontrarás todos los efectos que puedes aplicar como el de amplificar, inversión, ecualización eco, entre otros muchos que aparecen.

En el menú **Analizar** podrás realizar el análisis de elementos como contraste, espectro, entre otras acciones. Por último, aparece el menú **Ayuda**, que pone a tu disposición temas que te pueden ayudar a realizar algunas acciones o aclararte formas de proceder.

A continuación, se proponen algunas nociones elementales del trabajo con Audacity.

Práctica guiada 1

Imagina que estás creando una presentación de Power Point donde abordarás las ideas planteadas por Martí en sus cartas a María Mantilla y quieres en algunos momentos de esta presentación electrónica escuchar algunas de estas ideas esenciales.

Para esto deberás empezar por **gravar** estas ideas y comenzar por abrir la aplicación Audacity. Para ello debes seguir la siguiente ruta: **Botón inicio/todos los programas/Audacity**.

Con la aplicación abierta, empiezas por establecer la **calidad del sonido** que deseas obtener. Para ello, vas al menú **Editar/Preferencias**. En la ventana emergente que aparece encontrarás varias opciones, pulsarás sobre la opción **Calidad**.

El primer valor que encontrarás es la **Frecuencia de muestreo predeterminada**, la más importante, pues es la calidad con que vas a recoger el archivo. Podrás decidir entre las siguientes opciones (figura 1.110):

- 11 025 Hz: sonido de calidad inferior, pero de poco tamaño.
- 16 000 Hz: sonido de calidad algo mejor, aunque en algunos no se escucha la diferencia con el anterior.
- 22 050 Hz: sonido de calidad normal, aconsejable para cualquier grabación, en general.
- 44 100 Hz: sonido de alta calidad. Es el aconsejable para aplicaciones multimedia, en donde es importante la calidad del sonido.

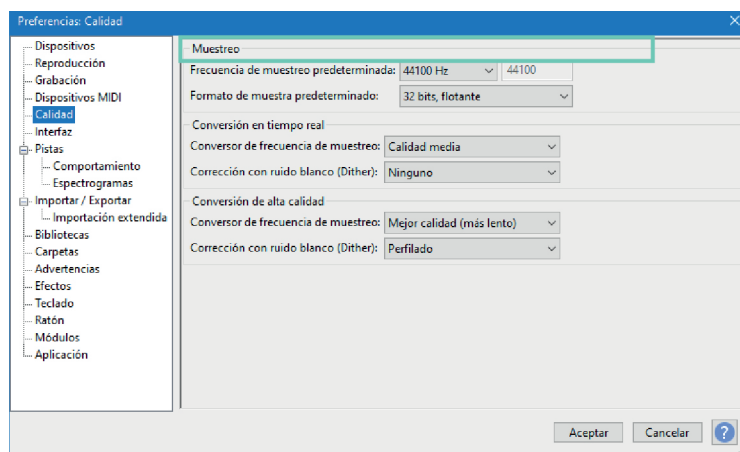


Fig. 1.110

Después de establecer las preferencias, darás clic en el botón **Aceptar**, de esta forma, ya podrás grabar tu voz. Para ello, se pulsa sobre el botón **Grabar** y comienzas a hablar. Cuando termines (y no antes) pulsarás el botón **Detener** (figura 1.111).

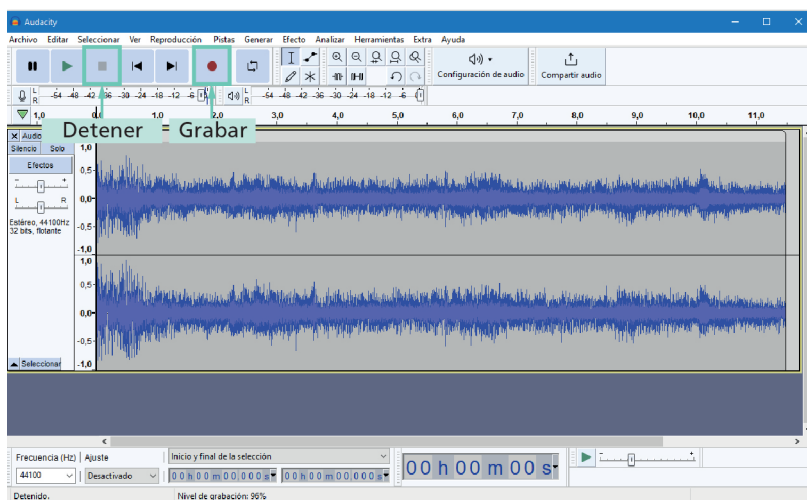


Fig. 1.111

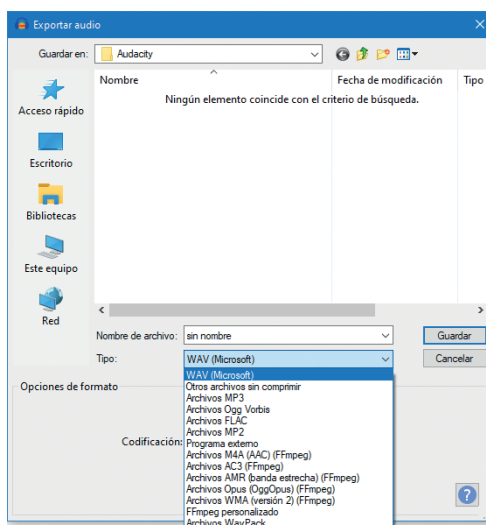
Observación

No debes decir nada antes de pulsar el botón **Parar**, pues pueden quedar “colas de sonido” que no deseas.

Luego podrás guardar el archivo grabado. Audacity tiene su tipo de archivo predeterminado, pero para poder usarlo en otro programa es necesario **exportarlo**. Para ello, se pulsa en el menú **Archivo/Exportar audio**.

En la ventana que aparece, se coloca el nombre que desees, el lugar donde vas a guardarlo y el tipo de archivo. En este caso seleccionarás WAV. Seguidamente darás clic en **Guardar** (figura 1.112).

Ahora bien, si quieres exportar a MP3, que no es más que un archivo WAV comprimido, seguirás el mismo procedimiento.



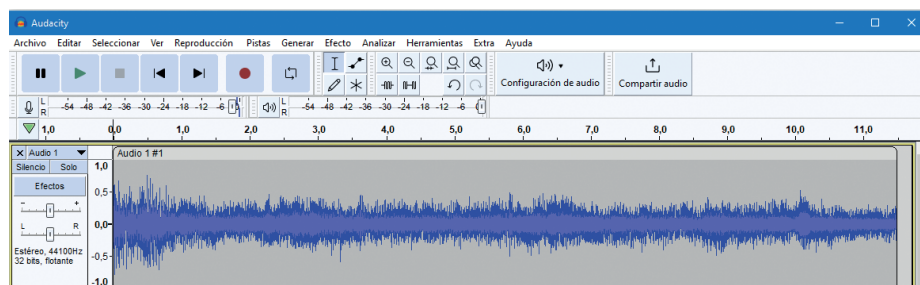


Fig. 1.113

Para realizar cualquier acción lo primero es seleccionar el fragmento que vas a editar y lo harás utilizando la herramienta **Selección**. Esta herramienta de forma predeterminada aparece seleccionada siempre. Con ella harás clic en el punto de inicio de la selección que deseas hacer y sin soltar, arrastrarás el puntero hasta el lugar donde concluirás la selección quedando de la siguiente forma (figura 1.114).

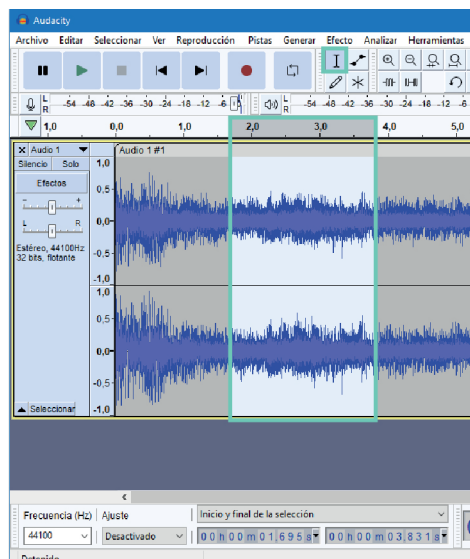


Fig. 1.114

Ahora bien, esta selección que has hecho la quieres cortar, ya que es un espacio de silencio, entonces vas al menú **Editar/Cortar**, así habrás eliminado el espacio de silencio. Es importante tener mucho cuidado en la selección del fragmento que vas a eliminar para que el corte no se note. Si el resultado no es el esperado siempre puedes deshacer la última acción dando clic en la opción **Deshacer** del menú **Editar**.

También podrás seleccionar fragmentos de otros archivos o del que estás editando para agregarlo en un lugar de nuestra pista, para ello usa-

Otras opciones importantes son el **Recortar**, que te permitirá eliminar todo lo que no se ha seleccionado en un archivo de sonido y la opción **silencio**, que elimina el fragmento seleccionado en un archivo, al sustituirlo con un silencio absoluto.

Actividad Práctica

1. Edita cada una de las grabaciones hechas por ti en la actividad práctica anterior y elimina las colas de sonido, espacios de silencio, teniendo en cuenta que no se deben notar los cortes.

Práctica guiada 3

Uno de los archivos grabados con alguna de las ideas que expresara nuestro Apóstol José Martí a María Mantilla tiene algunas partes con un volumen muy bajo que el resto de la grabación. Necesitas corregir esto y ponerlo a un volumen adecuado.

Para realizar esta acción deberás aplicar el efecto **Amplificar**, para lo cual comenzarás seleccionando el fragmento muy cuidadosamente, luego seleccionarás la opción **Amplificar** del menú **Efecto** (figura 1.115).

Podrás utilizar los valores que se te ofrecen de manera predeterminada; si no es así, podrás establecer los valores deseados arrastrando el botón deslizante hacia la derecha para subir e izquierda para bajar el volumen.

En el botón **Vista previa** se puede escuchar cómo queda el efecto para saber si está en el estado deseado antes de aplicarlo definitivamente. Si todo está correcto, pulsarás el botón **Aceptar** (figura 1.116).

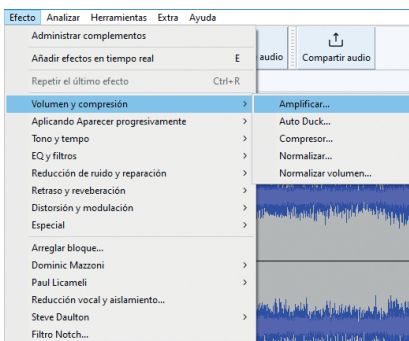


Fig. 1.115

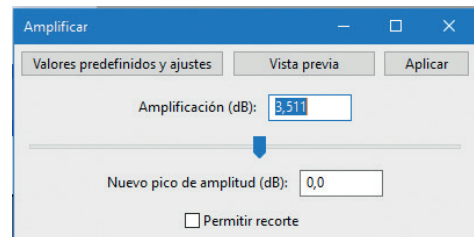


Fig. 1.116

Otro efecto que puedes aplicar para eliminar ciertos ruidos de fondo que aparecen en la grabación, bien debido al ambiente, o producidos por el micrófono o el ordenador, es el de **Reducción de ruido**.

Observación

Lo que hace el programa para eliminar el ruido es quitar determinadas frecuencias en las que se encuentra el ruido, pero junto con el ruido está eliminando partes de la grabación que comparten esas frecuencias. Por tanto, el uso de este efecto debe ser con mucha precaución.

Comenzarás seleccionando la parte donde detectes que hay más ruido. A continuación, pulsarás sobre el menú **Efecto** y elegirás la opción **Reducción de ruido**. Aparecerá una ventana emergente donde en primer lugar, vas a obtener el perfil de ruido que hay en la selección hecha, pulsando sobre el botón **Obtener perfil de ruido** (figura 1.117).

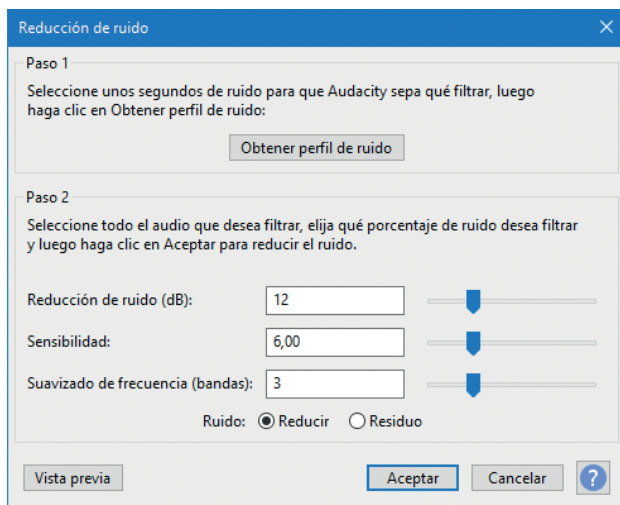


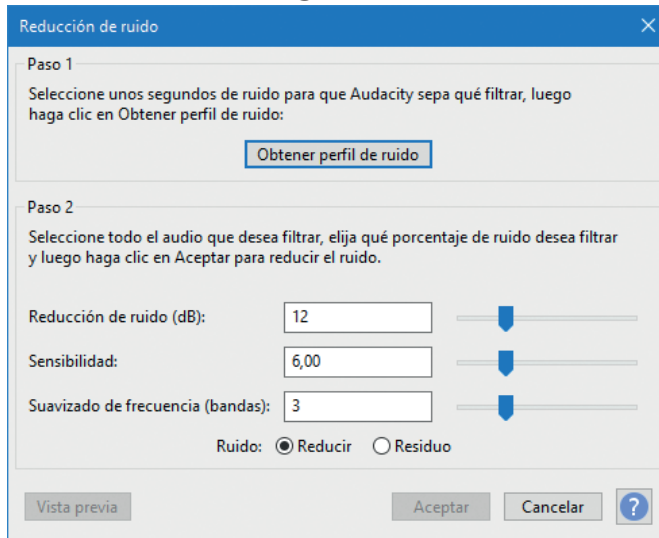
Fig. 1.117

Se cerrará esta ventana y seleccionarás entonces todo el audio que has grabado y volverás a ir al menú **Efecto**, donde vas a seleccionar la opción **Reducción de ruido**.

En la parte inferior de la ventana que aparece ajustarás los valores deseados moviendo los botones deslizantes y pudiendo comprobar el re-

sultado al dar clic sobre la opción **Vista previa**. Terminarás dando clic en **Aceptar** (figura 1.118).

Fig. 1.118



Observación

Al aumentar el valor del potenciómetro, el programa va añadiendo cada vez más frecuencias que se van a eliminar de nuestro archivo, lo que puede originar que el resultado sea un audio demasiado metalizado.

Podrás aplicar otros efectos de acuerdo a tus necesidades, el proceder deberá ser muy similar a los que has visto. Como ya te enseñamos los pasos básicos de la herramienta, ahora es tu turno para desarrollar tu creatividad.

Actividad práctica

1. Selecciona de los archivos de audio que grabaste tres de los que más te hayan marcado y aplica diferentes efectos para obtener archivos de óptima calidad.

Comprueba lo aprendido

1. Investiga y explica las diferencias fundamentales entre un códec y un formato de audio, entre los códec .mp3 y PCM, y entre los formatos de archivo .wav y .mp3.
2. Diga si es posible oír una pista de audio antes de seleccionarla para ser extraída. Explica cómo realizarías la extracción de alguna pista de sonido de un CD.
3. Encuentra los conceptos que corresponden a cada definición en esta sopa de palabras.

P	Q	T	G	F	E	Í	Í	Í	P	R
Í	Ñ	M	S	T	K	D	W	V	Ó	E
Ó	Ó	Y	H	Q	I	H	O	R	B	S
Z	M	M	I	Q	A	L	Ú	M	Á	O
S	U	A	M	P	L	I	T	U	D	L
N	E	Í	M	É	W	A	I	U	J	U
É	S	A	P	Á	O	L	U	X	U	C
Í	T	A	É	X	O	T	M	Z	Z	I
T	R	F	Ó	V	N	U	Á	X	N	Ó
D	E	C	W	F	G	R	Á	É	É	N
Í	O	B	I	T	R	A	T	E	N	Í

Definición

- Acción de tomar muestras a intervalos de tiempo regulares.
- Número que indica cuántos bits se utilizan para representar cada muestra.

- Número de vibraciones por segundo que da origen al sonido analógico.
 - Es la cantidad de fuerza o energía de un sonido.
 - Define la cantidad de espacio físico que ocupa un segundo de duración de ese audio.
4. Explica con tus palabras el procedimiento para aplicar un efecto de desvanecimiento a un archivo de audio en Audacity.
 5. Investiga y explica el procedimiento para realizar la mezcla de dos pistas de audio con Audacity.
 6. Busca algún dispositivo que te permita realizar grabaciones de audio y graba con tu voz al menos tres poemas de nuestro Apóstol José Martí, éditalos de forma tal que los unas en un solo archivo, como si fuera una sola grabación. Aplícale efectos y optimízalo, para publicarlo en la Web.

1.3 Tratamiento y edición de video

El video surge como una tecnología íntimamente ligada a la televisión, pues nació como auxiliar de esta para evitar que la programación fuera en directo, lo cual facilitó el trabajo de grabación, la planeación de horarios, el almacenaje de los programas y la reproducción de estos.

Etimológicamente, la palabra **video** proviene del latín *video, vidi, visum*; y significa “ver”. Esto de repente parece sencillo, pero adquiere cierto grado de complejidad al percibirse que el término video se utiliza con muy poca precisión, por lo que se presta a equívocos y generalizaciones diversas.

La tecnología de video permite manejar secuencias de imágenes que pueden ser señales de televisión o de otros dispositivos, así como de imágenes sintetizadas. Las secuencias de imágenes pueden ser capturadas y reproducidas a diferentes tasas, lo común en sistemas de cómputo son 30 cuadros por segundo.



Definición

El **video digital** es un tipo de sistema de grabación de video que trabaja utilizando en lugar de una señal analógica, una representación digital de la señal de video. Es una mezcla de audio, imagen, texto, animación, en un solo archivo digital.

A menudo el video digital se graba en cinta, para luego distribuirlo en discos ópticos, normalmente DVD. Hay excepciones, como las cámaras de video que graban directamente en DVD, las videocámaras de Digital 8, que codifican el video digital en cintas analógicas convencionales, y otras videocámaras que graban video digital en discos duros o memoria *flash*.

El video resulta de la exposición de imágenes o fotogramas, uno detrás de otro, en un tiempo determinado, y la cantidad de fotogramas por segundo que se muestran durante su reproducción, es un parámetro de su calidad. Por ejemplo, los DVD en Europa exhiben 25 fotogramas por segundo (25 fps).

El video digital tiene un tamaño dado por el ancho y por el alto, expresado en píxeles cuando se visualiza al 100 %, sin agrandar ni reducir; los estándares de video CD y DVD son 352 x 288 y 720 x 576 respectivamente. También tiene una proporción entre su ancho y su altura. Por ello es habitual una relación de 4:3 para los videos domésticos, mientras que en los DVD se suele trabajar con radios de 16:9. Al igual que el audio, tiene una velocidad de transmisión (bitrate) que define la cantidad del espacio físico (en bits) que ocupa un segundo de duración de ese video. Por tanto, el video tendrá más calidad cuanto mayor sea su bitrate y por eso, el archivo que lo contiene tendrá mayor peso.



Recuerda que...

Los **códec** son *software* que permiten comprimir los archivos de video y audio para que ocupen menos espacio en el disco. Estos han de ser capaces de descomprimir la información del archivo a la hora de reproducirlo. Teniendo en cuenta el **códec** empleado, varía el tiempo en la codificación y el tamaño resultante del archivo de video.

Los reproductores que se encuentran en las computadoras o dispositivos móviles pueden mostrar un video a pantalla completa o con una ampliación de 200 %, 300 %, etcétera. En estos casos, el video pierde calidad de imagen, que dependerá directamente del formato del archivo. Estos reproductores utilizan diferentes **códec** para reproducir los archivos de video, pues no todos están codificados de igual manera, cada uno tiene un **códec** específico que lo requiere para ser decodificado y reproducido. Por ejemplo, algunos de los **códec** más utilizados para el formato AVI son: DivX y XviD.

En la reproducción también interviene otro elemento importante, y es el sistema de televisión para el que está elaborado el video, pues de este elemento dependerá si se podrá ver en el televisor o no (figura 1.119). Estos estándares son:

- **NTSC (*National Television Estándar Comité* = Comité Nacional de Estándares de Televisión):** es utilizado en América del Norte, Centroamérica, Japón y otros países. En él cada fotograma está formado por 525 líneas y reproduce 30 fotogramas por segundo.
- **PAL (*Phase Alternation Line* = Línea Alternada en Fase):** es el más extendido en Europa y tiene 625 líneas por fotograma y reproduce 25 fotogramas por segundo.
- **SECAM (*Système Electronique Color Avec Memoire*):** utilizado en Francia y Rusia.
- **HDTV (*High-definition television*):** se define como dos veces mayor en resolución que la televisión convencional, con una relación de 16:9 de aspecto, una frecuencia de barrido de cuadro de 24 Hz o mayor y, por lo menos, tiene dos canales de audio de calidad CD.



Algo de historia

Las primeras películas de la historia fueron montadas a mano, fotograma a fotograma, por los propios directores. En 1924, la incorporación de una pequeña lupa en las moviolas facilitaba enormemente el trabajo.



Fig. 1.119

Los videos digitales se pueden guardar en distintos formatos, cada uno se corresponde con una extensión específica del archivo que almacena el video. Los formatos más utilizados son: AVI, MPEG, MP4, 3PG; WMV, MOV, RM; FLV (*Adobe Flash*, usado en Youtube, Google, etcétera).

- AVI (**A**udio **V**ideo **I**nterleaved = Audio y Video Intercalado): es el formato estándar para almacenar video digital. Cuando se captura el video desde una cámara digital, se suele almacenar en este formato con el **códec** DV (**D**igital **V**ideo = Video Digital), con muy buena calidad, pero muy pesado.
- El formato AVI puede ser visualizado en la mayoría de los reproductores: Windows Media, QuickTime, KMplayer, entre otros. Este admite distintos **códec** de compresión, los de mayor calidad de compresión y de una calidad aceptable son DivX y XviD, aunque suelen destinarse a películas de larga duración.
- MP4: es un formato de baja resolución pensado para videoconferencias.
- MPEG (**M**oving **P**ictures **E**xpert **G**roup = Grupo de Expertos de Películas): es un formato estándar para la compresión de video digital, son archivos de extensión MPG o MPEG. Admite distintos **códec** de compresión: MPEG-1 (calidad de CD), MPEG-2 (calidad DVD), MPEG-3 (orientado al audio MP3) y MPEG-4 (más orientado a la Web). Se puede reproducir con Windows Media Player y QuickTime, entre otros reproductores.
- WMV: ha sido desarrollado por Microsoft. Utiliza el **códec** MPEG-4 para la compresión de video. Solo se visualiza con una versión actualizada de Windows Media 7 o superior. Es ideal para publicar videos en Internet por razones de calidad y peso.

- MOV: es el formato de video de Apple con plataforma narrativa para Macintosh.



Saber más

Los archivos .wmv, .mov, .rm y .flv son los más adecuados para publicar un video en Internet por su adecuada relación calidad/peso y porque admiten **streaming**.

Los archivos AVI con **códec** de compresión baja son ideales para guardar los videos originales y los que tiene **códec** DivX o XviD son apropiados para videos de películas de cierta duración.

1.3.1 Edición del video digital

La edición de video es un proceso en el que se elabora un trabajo audiovisual o visual a partir de diferentes medios que pueden ser: archivos de video, fotografías, gráficos o animaciones, donde generalmente se incluye audio. Constantemente, la edición de video es realizada en una computadora, donde un programa especializado en edición de video permite al editor manipular los diferentes medios con ayuda de diferentes herramientas, para así lograr una composición y obtener un resultado final.

La edición básicamente consiste en poder reproducir los medios, recortarlos, ordenarlos, agregarles efectos, transiciones, filtros o textos, para mejorar o enfatizar el video.



¿Sabías que...?

Para hacer videos **no** necesitas ser un experto ni tener un equipo sofisticado, con una súper cámara digital y un gran **software**, ahora con una WebCam, una cámara digital sencilla o un teléfono móvil, podrás grabar, y la edición se convierte en un proceso sencillo con cualquiera de los programas que existen actualmente a tu disposición.

Cuando se habla de la edición de videos, se pueden mencionar dos modos:

- La **edición lineal**, que usa algún componente analógico, de modo que sobre la marcha se generan efectos de transición, montaje y superposición para crear el video final. Para este proceso hacen falta dos o tres magnetoscopios (uno de grabación).

- La **edición no lineal**, donde todo el soporte se digitaliza previamente, lo que permite editar lo que sea necesario sin necesidad de sincronización previa, retocar el resultado, etc., utilizando solamente la computadora en el proceso. Con un magnetoscopio puede ser suficiente, primero para digitalizar las tomas que se quieran procesar y finalmente, para registrar el resultado final.

Hay varias etapas en la edición de un video:

1. Una primera, donde se realiza la grabación teniendo en cuenta un grupo de reglas básicas como:
 - Conocer la cámara.
 - Contar con suficiente batería.
 - Tener limpia la óptica.
 - Buscar la máxima estabilidad en cada toma.
 - No abusar del **zoom**.
 - Llevar siempre la cámara preparada.
 - Cuidar la seguridad de la electrónica.
 - Máxima protección frente a golpes.
 - Debes grabar pensando en la edición.
 - Imaginar el lugar de la grabación.
 - Lugares en que no se debe grabar.
 - El sonido.
 - Durante la realización de las tomas: primero situar, después desarrollar planos comodín.
 - El encuadre.
2. Otra, realizada con la captura, donde debes conocer la fuente del video, si este es analógico o digital, si la cámara es USB; cómo realizarás el proceso de captura, el formato del video y los **códec** utilizados.

Para la captura o realización del video digital, se necesita una cámara de video digital o bien una cámara fotográfica o teléfono móvil que incorpore esta función. Debes tener en cuenta que los móviles usan diferentes formatos, por ello debes emplear algún programa para convertir los videos al formato que usa la aplicación que se utilizará para la edición.



Recuerda que...

- Las fuentes de video digital pueden ser variadas.
- La computadora y la cámara se conectan utilizando un cable IEEE-1394.
- La mayoría de las videocámaras DV y Digital 8 tienen un conector de cuatro contactos, mientras que las computadoras de sobremesa que vienen con puertos IEEE-1394 o con tarjetas IEEE-1394 instaladas, normalmente tienen conectores de seis contactos.
- En cambio, en las portátiles, al ser de cuatro contactos, el cable requerido será de cuatro contactos.

Los videos digitales, según donde vayan a ser empleados, también pueden optimizarse, para esto debes modificar algunos parámetros, como son:

- Los **códec** de compresión
- La resolución
- La velocidad de transmisión (**bitrate**)
- Las dimensiones
- La velocidad de los fotogramas
- El fotograma clave



Definición

El **fotograma clave**, es el fotograma completo que se guarda cada un tiempo determinado por la compresión.

Existen diversas herramientas para la **edición de video**, ya sean **open source** (código abierto) o **propietarios**; unas con más opciones que otras, pero todas de una forma u otra permiten editar el producto.

Ejemplos de editores de video para **software privativo** son:

- | | |
|----------------------|----------------------------|
| ■ Camtasia Studio | ■ Ulead Video Studio |
| ■ Sony Vegas | ■ Wondershare Video Studio |
| ■ Adobe Premiere Pro | |

En el caso de los editores **open source** (código abierto), se encuentran:

- | | |
|--------------|-------------------------|
| ■ Vidiot | ■ Kino GTK |
| ■ VirtualDub | ■ Lives |
| ■ Avidemux | ■ OpenShot Video Editor |
| ■ Cinelerra | ■ PiTiVi |
| ■ Kdenlive | ■ Lightworks |

En el desarrollo de este epígrafe, se usará el programa de edición de video digital Windows Movie Maker, que es un programa incluido como accesorio en todas las versiones del sistema operativo Windows, a partir de la versión ME (Mileniun). Este programa puede capturar señales de video tanto analógico como digital, aunque aquí solo se va a trabajar con video digital.

Se comenzará analizando su interfaz de trabajo. Para llegar a ella, se sigue la siguiente ruta: **botón Inicio/Todos los programas/ Windows Movie Maker**. Inmediatamente, aparecerá la pantalla principal de la aplicación con sus principales elementos. Esta pantalla puede variar en dependencia de la forma de visualización que desees, como **Guión gráfico** o como **Escala de tiempo** (figura 1.120).

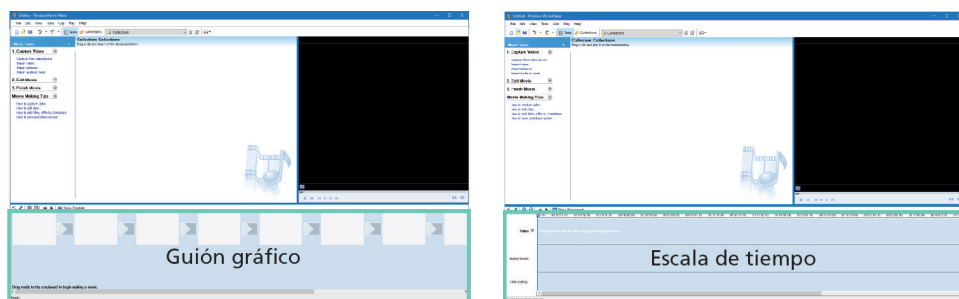


Fig. 1.120

En ambas interfaces encontrarás cuatro zonas bien diferenciadas, estas son (figura 1.121):

1. Tareas de película: listado en la parte izquierda, en donde encontrarás las herramientas para elaborar tu video. De manera general aparecen tres menús, que te permitirán realizar muchas acciones como: importar imágenes, transiciones de video, editar títulos, efectos y transiciones, etcétera.
2. Colección: aquí vas a importar tus imágenes y audios.
3. Un monitor: es una pequeña pantalla en fondo negro que te muestra una vista preliminar del video que estás realizando.
4. Pistas: se ubica en la parte inferior de la pantalla, es una sección gráfica en donde se van agregando todos los objetos y sonidos que formarán tu película.

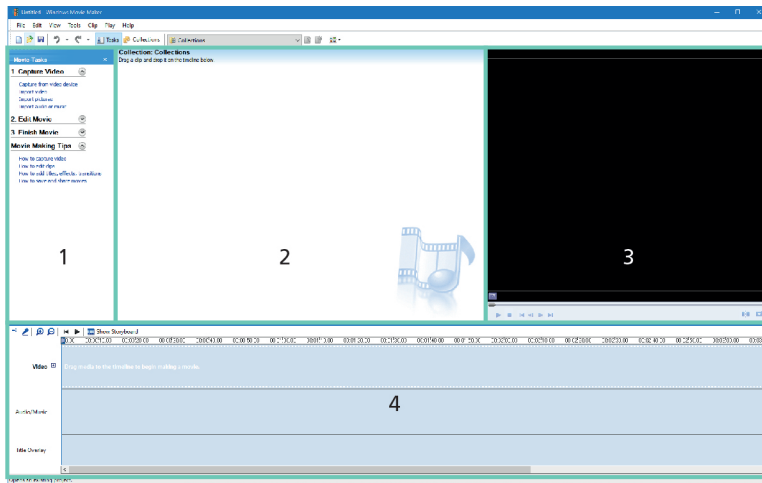


Fig. 1.21

Ya conociendo el entorno de trabajo, pasarás a crear los videos. Para eso, se te propone la realización de la siguiente práctica guiada.

Práctica guiada 1

Imagina que deseas elaborar un video que sirva como medio de enseñanza para el estudio de los símbolos patrios y atributos nacionales. Para esto, comenzarás por crear una carpeta con el nombre de tu proyecto y dentro colocarás todas las imágenes, videos y archivos de sonido que vas a utilizar en tu video. Luego, podrás comenzar.

Lo primero será importar todas las imágenes para tu video y lo harás dando clic sobre la opción **Importar imágenes** de la **zona Tareas de película**. Inmediatamente, aparecerá una ventana donde darás la ubicación de las imágenes que se van a importar, las seleccionarás y terminarás el procedimiento dando clic en **Importar** (figura 1.122).

Repetirás este procedimiento para importar el resto de los recursos, seleccionando por supuesto el tipo de recurso adecuado: **Importar video**, **Importar au-**

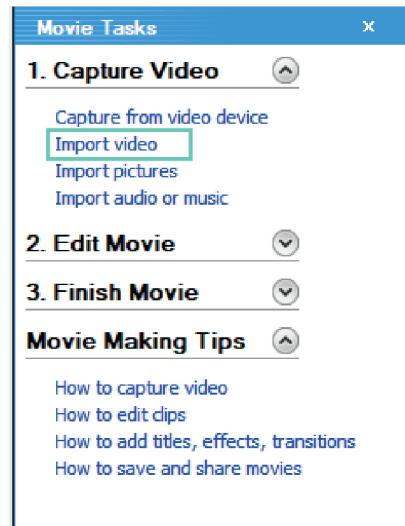


Fig. 1.122

odio o música o Capturar desde dispositivo de video. El resultado será visualizar en la **zona de las Colecciones** todos los archivos importados o capturados según su tipo (figura 1.123 a, b, c).



a



b. Archivo de audio



c. Archivo de imagen

Fig. 1.123

Cuando tengas todos los elementos multimedia que vas a utilizar, comenzarás a colocarlos en el proyecto. Para esto, los seleccionarás y arrastrarás hacia la línea de tiempo, según el orden de aparición de cada uno (figura 1.124).



Fig. 1.124

Todos los elementos que incluyas se irán colocando uno a continuación de otro, y dando clic sobre ellos podrás moverlos de lugar (figura 1.125).

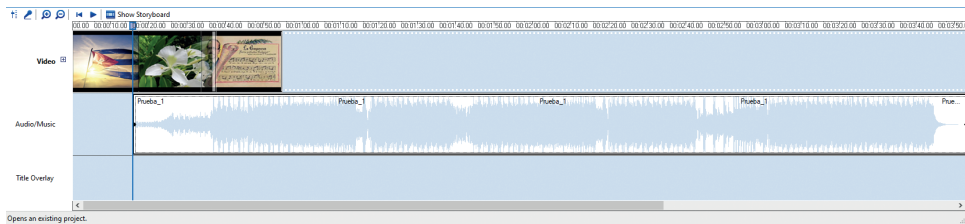


Fig. 1.125

En la línea de tiempo podrás realizar los ajustes necesarios para darle una duración a cada imagen arrastrando el cursor hacia la izquierda, para darle menor tiempo, y hacia la derecha, mayor tiempo o duración de vista (figura 1.126).

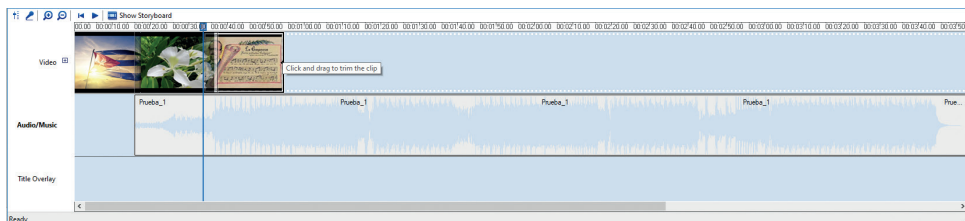


Fig. 1.126

De igual forma que en la imagen, se podrá ajustar la duración del audio o la música en la línea de tiempo correspondiente para hacerlo coincidir, por ejemplo, con la duración del video (figura 1.127).

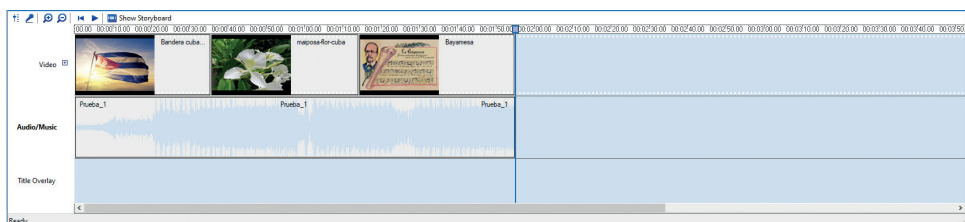


Fig. 1.127

También podrás realizar otros cambios referentes a la edición como crear títulos o créditos. Esto lo harás desplegando las opciones de **Editar película** que se encuentra en la **zona Áreas de película**, para luego seleccionar la opción **Crear títulos o créditos** (figura 1.128 a, b).

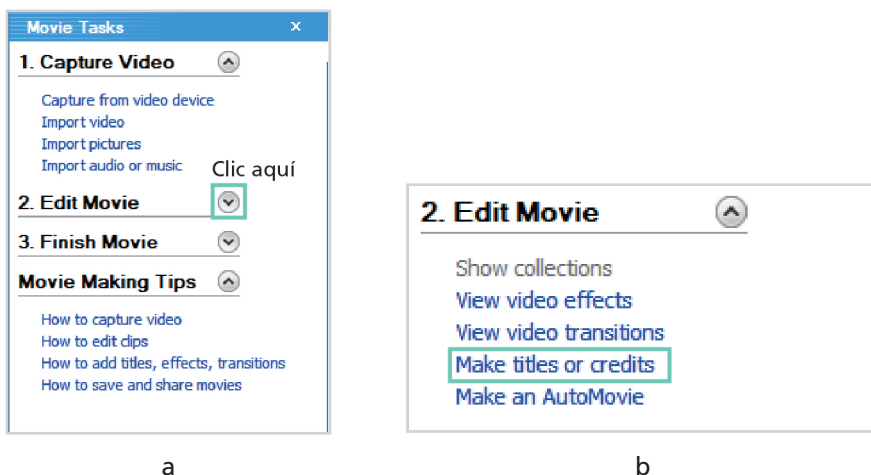


Fig. 1.128

Inmediatamente, aparecerá en el lugar donde estaba la **zona de Tareas de película** una ventana con varias opciones, donde seleccionarás lo que desees crear (figura 1.129).

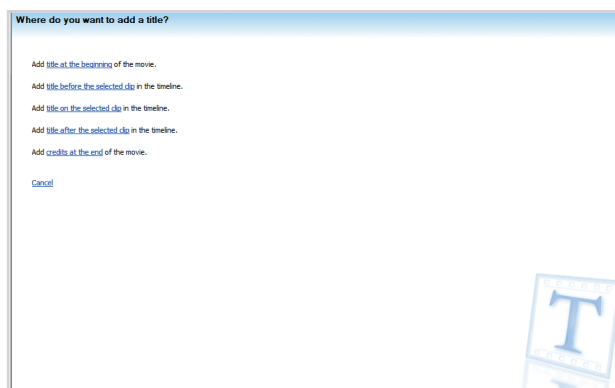


Fig. 1.129

En este caso, se seleccionará **Añadir título al principio de la película**, dando clic sobre esa opción. Aparecerá entonces la ventana, para que coloques el texto que desees agregar. También podrás cambiar la animación, el tipo de fuente y su color en las opciones que aparecen abajo (figura 1.130).

En la opción **Añadir créditos al final de la película**, podrás colocar tu nombre y los datos de tus colaboradores. Para finalizar, harás clic en listo, **Añadir título a la película**.

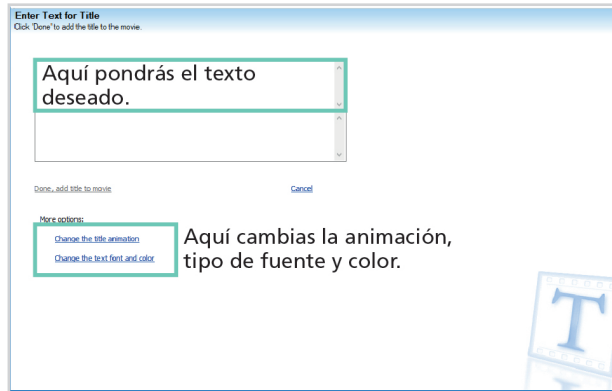


Fig. 1.130

A continuación, se agregarán algunas transiciones y efectos de video. Esto lo harás seleccionando la opción de **Editar película** que se encuentra en la **zona Áreas de película** y luego darás clic en la opción **Ver efectos de video** o **Ver transiciones de video** (figura 1.131 a, b).

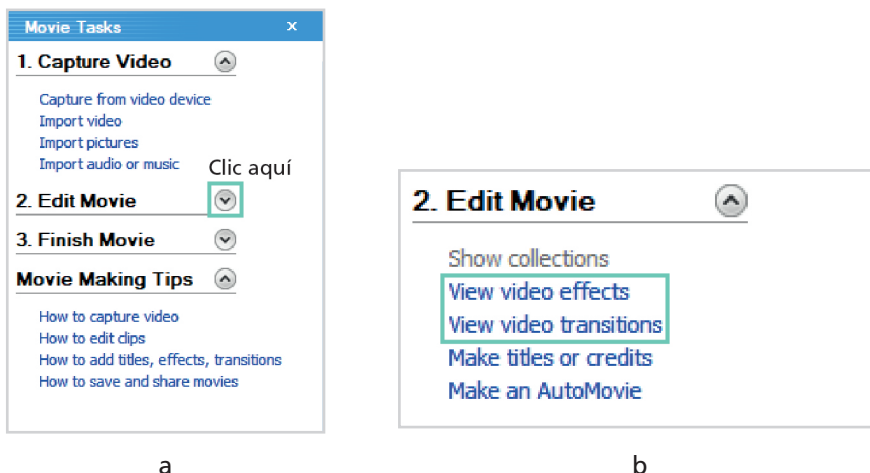


Fig. 1.131

Entonces, aparecerá en la **zona de Colecciones** todos los efectos de video o transiciones que se pueden agregar, como se muestra en las figuras 1.132 y 1.133.

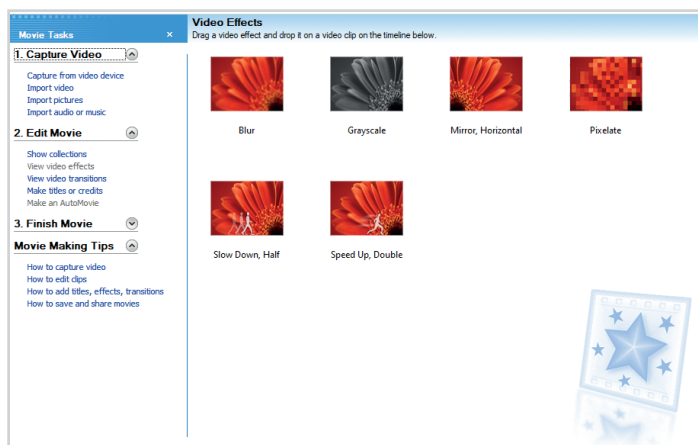


Fig. 1.132

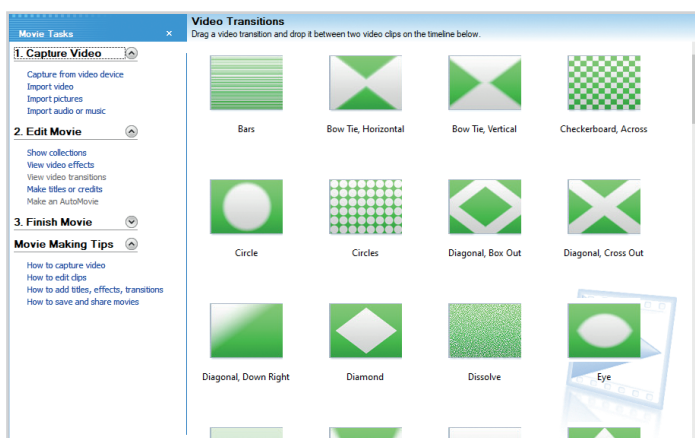


Fig. 1.133

Para aplicar cualquiera de las dos opciones, lo primero es seleccionar el efecto o transición que se va a usar y arrastrarlo hacia la línea de tiempo. En el caso del efecto de video, se arrastra hacia el elemento al que se le va a aplicar, y de esta manera podrás ver en ese elemento al que se le aplicó el efecto un símbolo a su derecha, como se muestra en la figura 1.134.

En el caso de la transición, se arrastra hacia la línea de transición, donde irán apareciendo cada una de las transiciones agregadas, como se muestra en la figura 1.135.

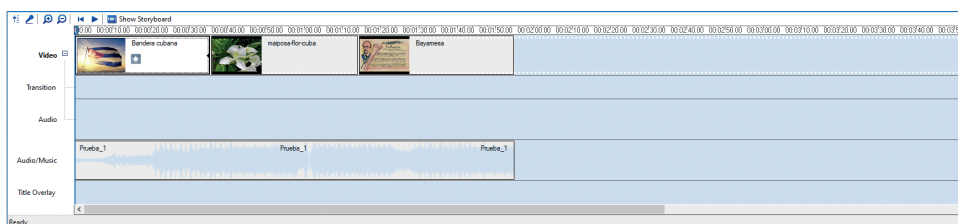


Fig. 1.134

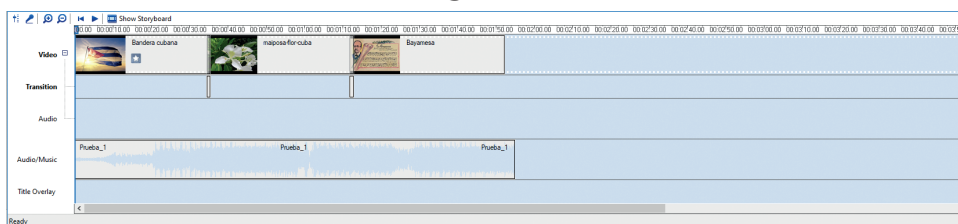


Fig. 1.135

Bueno, después de agregar todos los elementos multimedia deseados, aplicar los efectos de video, las transiciones, y agregar títulos y créditos, ha llegado el momento de grabar el video. Primero se guardará el proyecto, ya que de este modo si se necesitan realizar otros cambios al video este proyecto nos permitirá realizarlos; pues, si lo guardas como video, ya no se podrán hacer cambios. En el menú **Archivo** seleccionarás la opción **Guardar proyecto**. En la ventana que aparece, seleccionarás la ubicación.

Ahora, para la creación del video seleccionarás en el menú **Archivo** la opción **Guardar video**, y en la ventana que aparece el lugar donde deseas guardarlo, allí darás clic en **siguiente**. Lo próximo que harás será colocar el nombre y la ubicación del archivo que se generará en la ventana que aparece (figura 1.136).

Al dar clic en siguiente, podrás seleccionar la calidad del video y con esta el tamaño del archivo. A continuación, comenzará el proceso de creación del video y en la ventana se podrá apreciar el nombre del archivo donde se guardará y a qué por ciento va la creación, como se puede apreciar en la figura 1.137.

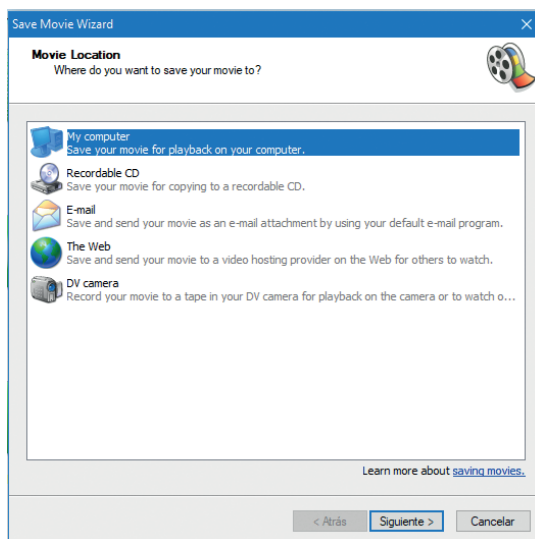


Fig. 1.136

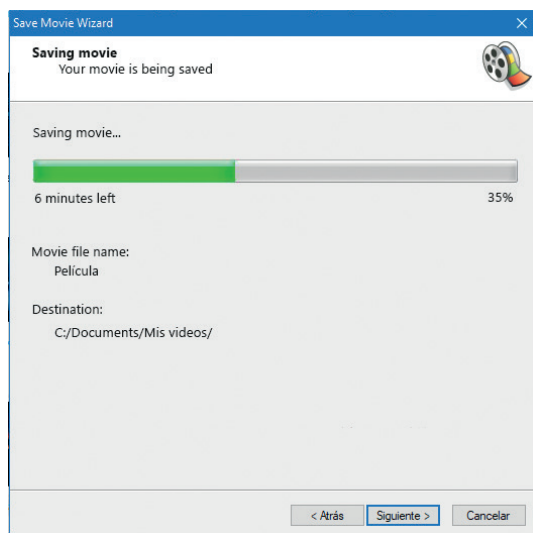


Fig. 1.137

Al concluir este proceso, aparecerá una nueva ventana para que selecciones, si lo deseas, la opción de **Reproducir el video** al finalizar la grabación. Con este paso, ya tendrás el video terminado.

Actividad práctica

1. Recopila imágenes, videos, audios u otros materiales multimedia digitales que te permitan realizar un video sobre la vida de nuestro Comandante en Jefe Fidel Castro.

Comprueba lo aprendido

1. Explica con tus palabras la diferencia entre guardar un proyecto y una película en Windows Movie Maker.
2. Explica cómo insertas imágenes fijas en un proyecto de Windows Movie Maker.
3. Realiza una búsqueda de materiales multimedia relacionados con las adicciones y sus consecuencias. Elabora un video informativo para presentar a tus compañeros sobre este tema.
4. Investiga a qué se denomina “obsolescencia programada”, recopila materiales multimedia y elabora un video sobre este tema, para que lo enseñes a tus compañeros de grupo.



CAPÍTULO 2

Introducción a la programación orientada a objetos y conducida por eventos

Hace poco tiempo las computadoras de escritorio se volvieron una herramienta esencial para la industria y las actividades diarias de cada persona, incluso en los años 90 del siglo xx, el poder de las computadoras personales aumentó de manera radical. Únicamente las personas más visionarias imaginaron la influencia que la computación tendría en la sociedad, pues este desarrollo tecnológico ha permitido que los dispositivos móviles también tengan el poder de procesamiento de una computadora de escritorio y puedan conectarse entre sí por medio de la *Internet*.

Es por ello que en la era de la economía del conocimiento, las personas y los países que mayor desarrollo alcanzan son los que remplazan el trabajo manual por el trabajo intelectual, ya que el conocimiento se utiliza para generar valor y riqueza, al transformarlo en información. Por ejemplo, el valor de Facebook es mayor que millones de productos manufacturados.

En la actualidad, la capacidad de innovación e imaginación de los pensadores de la computación ha permitido la decodificación del genoma humano, el desarrollo de sistemas de información para alertar sobre condiciones climáticas peligrosas y la traducción de lenguajes antiguos, entre otras muchas soluciones a problemas importantes de la sociedad.

Si analizas bien las iniciativas tecnológicas que favorecen a prácticamente todas las áreas de estudio y del quehacer humano, podrás percibir que estas son creadas por personas creativas que desarrollan habilidades superiores del pensamiento, como el razonamiento abstracto, el pensamiento crítico, la resolución de problemas y la toma de decisiones, entre otras habilidades. De ahí la importancia del desarrollo del Pensamiento

computacional, que al igual que el conocimiento del idioma o la aritmética, será una habilidad y una actitud de aplicación mundial para que todas las personas se encuentren orientadas a la creatividad y a la innovación.

2.1 El Pensamiento computacional

Antes de definir el concepto de *Pensamiento computacional*, observa por un momento lo que te rodea y describe mentalmente las situaciones en las que una computadora o cualquier otro dispositivo electrónico te han ayudado en la realización de alguna tarea, ya sea porque los has utilizado para comunicarte o porque te han proporcionado información.

Estamos seguros de que probablemente pensaste en un teléfono inteligente, que te da la hora, te permite hacer cálculos, te ofrece entretenimiento, te permite leer libros electrónico y que es tu dispositivo de preferencia para comunicarte con tus amigos, además este tiene el potencial para instalarle cientos de las aplicaciones disponibles; tal vez observaste una cámara digital, o recordaste a un amigo que tiene una pulsera electrónica que mide su actividad física, un reproductor de música, un dispositivo para visualizar películas y series de televisión bajo demanda, entre muchas otras cosas.

Ya luego de esta observación, puedes afirmar con certeza que las computadoras y las tecnologías de información están en todas partes y que nos han permitido extender las capacidades del pensamiento humano para realizar actividades y resolver problemas de una manera más rápida y eficiente.

Ahora reflexiona, ¿qué crees que ocurre cuando vas a una tienda y la cajera pasa los artículos por un lector de códigos de barras y al final te da el total a pagar?, ¿de qué manera es posible que un sensor de movimiento haga que la puerta de un aeropuerto se abra cuando detecta a una persona? y ¿cómo se pueden tener ahorros significativos de energía eléctrica? Los ejemplos mencionados con anterioridad te invitan a conocer un poco más sobre cómo funcionan los sistemas informáticos y de qué manera se puede aprovechar su potencial para ayudarte a resolver problemas o situaciones cotidianas.

¿Sabías que...?

Se llama **domótica** a los **sistemas** capaces de **automatizar** una vivienda o edificación de cualquier tipo, aportando servicios de gestión energética, seguridad, bienestar y comunicación, los que pueden estar integrados por medio de redes interiores y exteriores de comunicación, cableadas o inalámbricas, cuyo control se puede realizar desde dentro y fuera del hogar. Se podría definir como la integración de la tecnología en el diseño inteligente de un recinto cerrado.

Aunque existen diversas definiciones de Pensamiento computacional, se plantea que este es un proceso para la solución de problemas a través de diferentes técnicas como: organización y análisis de la información de manera lógica; representación de la información mediante abstracciones tales como modelaciones, simulaciones o diagramas; formulación de problemas que te permitan usar una computadora y otras herramientas para resolverlos; automatización de soluciones mediante el pensamiento algorítmico (una serie ordenada de pasos); identificación, análisis e implementación de posibles soluciones con el objetivo de lograr la combinación más efectiva y eficiente de pasos y recursos; así como la generalización y transferencia de este proceso de resolución a una amplia variedad de problemas.

Es decir, el **Pensamiento computacional** es una metodología basada en la implementación de los conceptos básicos de las ciencias de la computación para resolver problemas cotidianos, diseñar sistemas domésticos y realizar tareas rutinarias. Esta nueva forma de abordar los problemas nos permite resolver con eficacia y éxito, las situaciones que de otra forma no son tratables por una persona. Es un enfoque que no reemplaza el énfasis en creatividad, razonamiento o pensamiento crítico, pero refuerza esas habilidades al tiempo que realza formas de organizar el problema de manera que el ordenador pueda ayudar. A continuación se te darán más detalles de algunas de estas técnicas empleadas en este proceso.

Abstracción

El término **abstracción** se entiende popularmente en algunas ocasiones con un significado contrario al que tiene en realidad. Cuando algo parece o es de difícil entendimiento (por ejemplo, una obra de arte), se suele expresar “es muy abstracto”. Sin embargo, por el contrario, la abs-

tracción es el filtro utilizado para quedarse con lo que se considera esencial, eliminando toda la complejidad innecesaria. Se refiere al proceso de eliminar lo que no nos sirve o la parte compleja de un problema que estamos analizando. Dicho de otra manera, abstraer nos permite generalizar y nos invita a observar únicamente la parte más esencial del problema (figura 2.1).



Fig. 2.1

El concepto de abstracción es una de las principales ideas del Pensamiento computacional en el proceso de solución de problemas, ya que este tipo de pensamiento promueve el desarrollo de las habilidades de abstracción a través del análisis, el diseño y el modelado de soluciones por medio de algoritmos, programas, estructuras de datos abstractos, etcétera.

Para resolver un problema, normalmente definimos una abstracción de acuerdo con nuestros intereses. A veces el nivel de abstracción que se define está restringido por las limitaciones que se tienen en nuestros sistemas. Un ejemplo claro en donde se usa la abstracción es en los videojuegos, donde cada elemento que se grafica es una abstracción que el analista diseñó para representar determinados objetos.

Debes saber que la abstracción puede incluir cambios importantes en la formulación de un problema, pero que facilitan su solución y nos puede llevar a soluciones novedosas.

Ejemplo

José decidió jugar un juego sencillo de aritmética con Marta. El juego es el siguiente: hay dos jugadores que tienen que seleccionar de manera alternada un dígito entre el 1 y el 9 en tres turnos. Una vez elegido un dígito, este no se puede volver a seleccionar. El jugador que logre que sus dígitos sumen 15 al tercer intento, gana.

Marta se quedó pensativa un tiempo, tomó papel y lápiz y escribió algo. Después le dijo a José que estaba lista para jugar. José había jugado antes y pensó que sería fácil ganarle a Marta. Sin em-

4	9	2	15
3	5	7	15
8	1	6	15
15	15	15	15

Fig. 2.2

bargo, la mayor parte del tiempo José perdía o ninguno lograba tener tres dígitos que sumaran 15. Cuando José le preguntó a Marta qué tenía en el papel que consultaba todo el tiempo durante el juego, Marta le enseñó un cuadro mágico. Un cuadro mágico es aquel en donde la suma de todas las columnas, filas y diagonales, es lo mismo (figura 2.2).

Marta le explicó que cada vez que él seleccionaba un número le ponía una cruz al cuadro mágico, y cada vez que ella seleccionaba un número le ponía un círculo al cuadro mágico. Para ganar le dijo que todo lo que tenía que hacer era jugar “gato” en el cuadro mágico. Lo que hizo Marta fue abstraer lo más importante del juego: seleccionar 3 dígitos diferentes que sumaran 15 y pensar en cómo representarlo para que fuera más fácil jugar.

Descomposición

Dependiendo de la magnitud de un problema, este puede descomponerse en partes más pequeñas o manejables. Es decir, un planteamiento complejo es, regularmente, el resultado de otros menores; por ejemplo, el tráfico en una ciudad grande se puede descomponer desde diferentes perspectivas como: la educación vial, la cantidad de automóviles circulando, las vialidades, los señalamientos, el transporte urbano, las áreas para otros medios como bicicletas, etc. Generalmente, si resuelves de manera independiente problemas pequeños, podrás resolver uno más grande (figura 2.3).

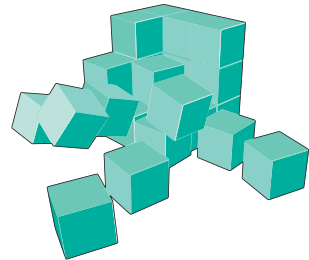


Fig. 2.3

Simulación

Juega un rol muy importante dentro de las características del Pensamiento computacional. Es a través de simulaciones que se puede anticipar el resultado de un problema. Un ejemplo de estas es el que utilizan los pilotos antes de volar un avión real, pues mediante una simulación controlan las variables de velocidad, altitud, temperatura, viento, dirección, etc.; de tal mane-



Fig. 2.4

ra que al modificar cualquiera de los parámetros, la forma de pilotear puede cambiar dramáticamente (figura 2.4).

Las personas que desarrollan estas técnicas basadas en el ordenador están en disposición de resolver problemas complejos no solo por sacar provecho de la potencia computacional de los ordenadores, sino también por la capacidad de los lenguajes del ordenador en describir sistemáticamente un problema en varias capas de abstracción y de describir el interface entre dichas capas sin ambigüedad. Esta habilidad aumenta de forma absoluta la complejidad de los problemas reales, para los cuales se puede encontrar una solución buena y eficiente.

2.1.1 Definición de problema

Cada día te enfrentas a situaciones como decidir qué desayunar, observar tu agenda para priorizar tus actividades, determinar la ruta para ir al instituto o al trabajo, asistir puntualmente a clases, comer, descansar, divertirse y volver a empezar. Siempre tomas decisiones basadas en la experiencia y el conocimiento; algunas tareas son fáciles y otras no tanto, pero, ¿cuándo se convierten en problemas? (figura 2.5).

La Real Academia Española (RAE) define **problema** como el “conjunto de circunstancias o hechos que dificultan la consecución de algún fin” (2015), según el diccionario Oxford un **problema** es un “asunto o situación no bienvenida o placentera, dañina, y que debe afrontarse y resolverse” (2015). Entonces tomando como referencia lo expuesto anteriormente, podemos crear una definición y asegurar que un problema es una situación que amerita una serie de acciones para pasar de un estado actual no deseado, a un estado adecuado o deseado. Los problemas pueden ser de diferentes tipos: académicos, financieros, sociales, de salud, de relaciones interpersonales, familiares, psicológicos, administrativos, etc. Cada uno de estos se afrontan de manera individual y para resolverlos evocamos a nuestros conocimientos y habilidades personales. Por ejemplo:

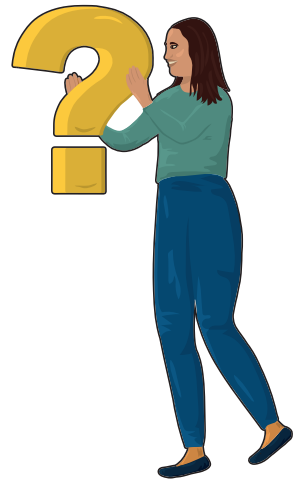


Fig. 2.5

Problema: Luis ha tenido un desempeño académico bajo en la materia de Matemáticas en el último mes y desea mejorar su rendimiento. Considerando una de las definiciones anteriores, se puede afirmar que, para resolver la situación planteada anteriormente, Luis debe realizar una serie de acciones como estudiar, hacer tareas, prepararse para los siguientes exámenes, etc.; de esta manera, él podrá superar su condición actual y transformarla en una ideal que refleje un mejor desempeño académico.

A continuación, se describirán algunas estrategias y metodologías que nos ayudarán a entender las fases o etapas por las que pasa una problemática.

2.1.1.1 Metodología para definir y resolver un problema

Aunque no hay un método único, algunos autores concuerdan en la utilización de algunas técnicas aplicables a diferentes áreas: matemáticas, física, psicología, sociología, entre otras. Entre dichas técnicas o estrategias se encuentran las siguientes: **ensayo y error**, **abstracción**, **descomposición**, **desarrollo de algoritmos**, **lluvia de ideas**, entre otras. A estas se les pueden aplicar las **metodologías** que se presentan a continuación:

Metodología general

En la **metodología general** la solución de problemas pasa por cuatro etapas principales que son (esquema 2.1):

1. Etapa de recolección de información: entender el problema es esencial para resolverlo, por lo que se requiere hacer una investigación extensa acerca de este: qué es lo que se desea obtener, qué se necesita, qué obstáculos se pueden presentar. Un factor clave para esta fase es la habilidad de discriminación de la información para diferenciar lo que es relevante y lo que no lo es, en la que interviene un proceso de **abstracción** (figura 2.6).



Fig. 2.6

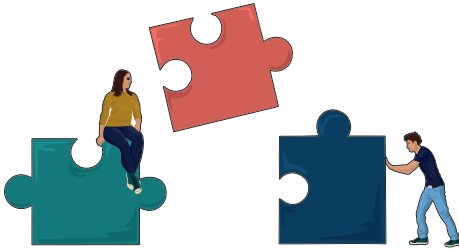


Fig. 2.7

período es la de **ensayo y error**, en la que se intenta una solución y si no funciona, se prueba una segunda hasta resolver la situación (figura 2.7).

3. Etapa de implementación de la solución: esta comienza con la toma de decisiones; una vez que se han propuesto algunas alternativas, es necesario llevar a cabo una de ellas. En algunas situaciones, el problema se puede resolver de más de una forma; sin embargo, algunas soluciones pueden ser mejores que otras porque llevan más o menos tiempo o pueden ser más fáciles o eficientes al momento de aplicarse. Esta fase implica llevar a cabo un plan de acción (figura 2.8).

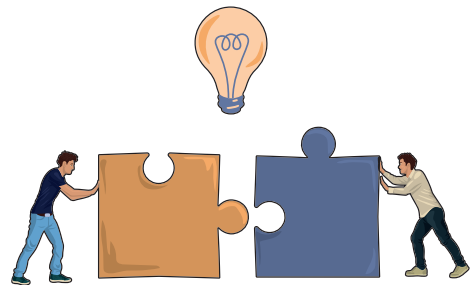


Fig. 2.8



Fig. 2.9

4. Etapa de evaluación: es el último peldaño de esta metodología. Una vez que la solución o plan de acción ha sido implementado, se necesita considerar si el objetivo principal se ha cumplido, si no es así, es necesario considerar otra alternativa o plan de acción. Para esto, se debe revisar cada acción y, si es necesario, retomar la etapa dos: generación de soluciones (figura 2.9).



Esquema 2.1 Etapas de la metodología general para resolver un problema.

Se verán a continuación dos ejemplos de cómo utilizar la metodología general en la solución de problemas:

Ejemplo 1

El papá de Jorge le ha ofrecido para su cumpleaños comprar una consola nueva para jugar. Jorge no está seguro de cuál adquirir y desea que le ayudes a tomar la mejor decisión.

1. **Recolección de información:** lo primero que debes hacer es concentrar toda la información posible acerca de las consolas que se venden actualmente: precio, variedad de juegos, capacidad para jugar en línea, accesorios, características de *hardware*, etcétera.
2. **Generación de soluciones:** al hacer un contraste de las características de cada una de ellas, se han reducido las opciones de compra a solamente dos consolas que cumplen, en apariencia, con las necesidades y gustos propios de Jorge.
3. **Implementación de la solución:** ahora sabes que tienes una semana para probar los productos y si no estás satisfecho con tu compra, puedes solicitar la devolución de tu dinero o cambiarlo por otro. Dado lo anterior, has decidido comprar la primera opción para Jorge.
4. **Evaluación:** después de unos días, te das cuenta que la compra no fue la mejor y has decidido regresar a la tienda y cambiarla por la otra consola que habían contemplado.

Ejemplo 2

Estás por terminar tus estudios preuniversitarios y deseas convertirte en ingeniero en Informática, pero no estás seguro de qué pasos necesitas seguir para cumplir con tu objetivo.

1. **Recolección de información:** lo primero que necesitas hacer es conocer el plan de estudios de esta carrera en diferentes universidades y saber qué es lo que hace un ingeniero en sistemas; también debes acudir al centro de consejería de tu escuela para informarte acerca de costos, becas, tiempo para terminar, mercado laboral, etcétera.
2. **Generación e implementación de la solución (etapas 2 y 3):** ahora debes prepararte para el examen de admisión a diferentes escuelas y diseñar el plan de estudios que debes cubrir asegurándote de cumplir con los requisitos de la carrera.

3. Evaluación: si el examen de admisión es acreditado, la solución va en camino, pero si no, se deben tomar acciones como volver a intentarlo, si el resultado no está muy distante de los requerimientos, aplicar en otra universidad o inclusive, cambiar de objetivo principal, que en este caso, sería cambiar de carrera a otra que pudiera ser afín con lo que desees.

Como te habrás dado cuenta, en los ejemplos anteriores se llevó a cabo la metodología propuesta cumpliendo con las cuatro etapas mencionadas y de esta manera fue posible solucionar una problemática cotidiana. Ahora pasarás a estudiar otra metodología.

Metodología de Polya

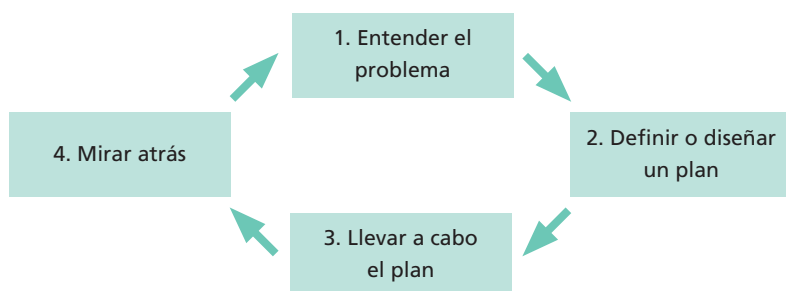
George Polya, en su libro *Cómo resolverlo* (1945), propone un método de cuatro pasos para la resolución de problemas matemáticos:

1. Entender el problema.
2. Definir o diseñar un plan.
3. Llevar a cabo el plan.
4. Mirar atrás.

A continuación, se te describen las características de cada uno estos pasos (esquema 2.2):

1. Entender el problema: para esto debes atender las siguientes interrogantes: ¿puedes describir el problema con tus palabras?, ¿qué tratas de hacer o encontrar?, ¿qué es lo que se desconoce?, ¿qué información obtienes del problema?, ¿qué información falta?, ¿qué información no se necesita del todo?
2. Definir o diseñar un plan: busca un patrón, examina situaciones relacionadas y determina si la misma técnica puede ser aplicada; examina un caso especial o más simple para comprender el problema original, haz una tabla, un diagrama; escribe una ecuación, usa prueba y error, trabaja hacia atrás e identifica una submeta.
3. Llevar a cabo el plan: implementa la estrategia o estrategias planteadas en el paso 2, y ejecuta cualquier operación o acción necesarias. Revisa cada paso según como vas avanzando, esto puede ser intuitivo o de manera formal, y lleva un registro exacto de tu trabajo.

4. Mirar atrás: revisa los resultados del problema original y, de ser necesario, consigue una prueba e interpreta la solución en estos términos: ¿tu respuesta es coherente y razonable?; determina si hay manera de encontrar otra y, si es posible, determina problemas más generales o relacionados en los que puedas aplicar esta técnica.



Esquema 2.2 Pasos de la metodología de Polya para resolver un problema.

Ahora verás a partir del análisis de un ejemplo la metodología de Polya utilizando como plan de solución *el ensayo y error* y *el planteamiento de una ecuación*.

Ejemplo (ensayo y error/escribir una ecuación)

Un par de hermanos estuvieron ahorrando una cantidad de dinero cada mes durante un año, al final entre los dos lograron obtener 192 pesos. Si el hermano mayor ahorró el triple que el menor ¿cuánto ahorró cada uno de ellos?:

1. Entender el problema: usa las preguntas del paso 1 con sus respectivas respuestas (ver tabla 2.1).

Tabla 2.1

Preguntas	Respuestas
¿Qué tratas de hacer o encontrar?	Una cantidad de dinero
¿Qué es lo que se desconoce?	Cuánto ahorró cada uno de los hermanos
¿Qué información tienes del problema?	Sé que entre los dos hermanos ahorraron 192 pesos y que el mayor ahorró el triple que el menor

2. Diseñar un plan: plantea dos soluciones, la primera está basada en prueba y error, mediante la creación de una tabla, y la segunda será definir una ecuación.
3. Llevar a cabo el plan:
 - Solución utilizando **Ensayo y error** (ver tabla 2.2)
 - Podrías decir que el hermano menor ahorró 40 pesos. Dado que el mayor juntó tres veces más, entonces multiplica 40×3 (120 pesos). En este caso, el total sería de 160 pesos, lo cual es menos de los 192 pesos que consiguieron.
 - Podrías decir que el hermano menor ahorró 50 pesos. Dado que el mayor reunió tres veces más que el menor, entonces multiplica 50×3 (150 pesos). En este caso, la suma total es de 200 pesos, lo cual excede la cantidad ahorrada.
 - Podrías decir que el hermano menor ahorró 48 pesos. Dado que el mayor ha ahorrado tres veces más que el menor, entonces multiplica 48×3 (144 pesos). En este caso, la suma entre los dos da como resultado la cantidad correcta.

Tabla 2.2

Ahorro del hermano menor	Ahorro del hermano mayor	Ahorro total	Solución
40	120	160	Menos de lo que ahorraron
50	150	200	Más de lo que ahorraron
48	144	192	Justo lo que ahorraron

- Solución utilizando **Definir una ecuación**

Supón que “x” es la cantidad que ahorró el hermano menor. Si el mayor reunió tres veces más y esta suma da un total de 192 pesos, se puede expresar todo de la siguiente manera: $3x + x = 192$

Resuelve la ecuación:

$$3x + x = 192$$

$$4x = 192$$

$$x = \frac{192}{4}$$

$$x = 48$$

Como podrás ver, en la tabla de *prueba y error* encontramos que el hermano menor sí ahorró 48 pesos, por lo tanto, nuestra ecuación es correcta.

4. Mirar atrás: una vez que resolviste el problema puedes constatar que la solución es lógica y que sí atendiste la necesidad original, pues tanto la tabla como la solución a la ecuación dieron el mismo resultado. Observa cómo este tipo de soluciones pueden ser aplicadas a otros problemas similares.

Ahora vas a analizar con el siguiente ejemplo la metodología de Polya utilizando como plan de solución **hacer un diagrama**.

Ejemplo

En una carrera de coches, los primeros cinco autos en llegar en el siguiente orden fueron: rojo, verde, azul, blanco y negro. Considera que:

- El rojo llegó siete segundos antes que el negro.
- El azul llegó seis segundos después que el blanco.
- El verde llegó ocho segundos después que el blanco.
- El negro llegó dos segundos antes que el azul.

Resuelve la pregunta: ¿en qué orden llegó cada uno de ellos?

1. Entender el problema: quieres saber el orden en que llegaron cinco automóviles en una carrera y ya conoces algunos datos de la relación de tiempo que hay entre ellos.
2. Diseñar un plan: utiliza un diagrama que muestre la llegada de cada uno de los autos.
3. Llevar a cabo el plan: realiza un diagrama que muestre la llegada de cada uno de los coches. Aquí puedes hacer marcas que señalen la diferencia en segundos de cada uno (figura 2.10).

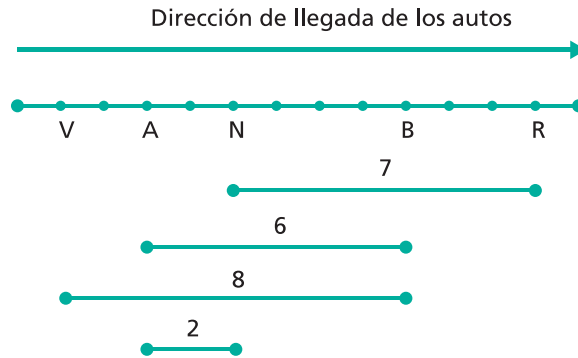


Fig. 2.10

4. Mirar atrás: en este caso podrás observar, no solamente el orden de llegada de los autos, sino también el tiempo que hubo entre cada uno de ellos de acuerdo con la información proporcionada.

Como habrás podido apreciar las metodologías estudiadas te servirán para resolver cualquier tipo de problemas, así que ahora se proponen un conjunto de actividades para que puedas consolidar tus conocimientos.

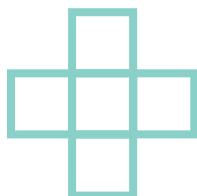
Comprueba lo aprendido

1. Analiza las técnicas que componen la metodología del **Pensamiento computacional** para resolver problemas y las técnicas de la metodología de Polya. Elabora una tabla comparativa donde expresas las similitudes y diferencias de estas metodologías.
2. Luisa asistirá a un concierto a las 4 p.m. de la tarde, pero tiene una clase que acaba a las 3 p.m. Ella quiere saber en qué horario debe tomar el autobús, de tal manera que llegue a tiempo y no tenga que salir temprano de la clase. Si los autobuses salen cada 15 min (10:00, 10:15, 10:30, etc.) y el trayecto es de 30 min, ¿cuál es el mejor horario para tomar el autobús? ¿Cuál fue el razonamiento que seguiste? ¿Cuáles fueron las entradas, salidas y el problema por resolver?
3. Las vacaciones de verano están a punto de terminar. Daniel se da cuenta de que no tiene una mochila para llevar los libros a la escuela, así que corre a la tienda y compra la mochila que le pareció

más bonita, con esto el problema parecía resuelto; sin embargo, el primer día de clases, Daniel se da cuenta de que los libros no caben en la mochila nueva. ¿Por qué la manera de resolver el problema no fue la mejor? ¿Qué le faltó a Daniel? ¿Tú cómo lo hubieras resuelto?

4. Ana y Raúl están organizando una fiesta para festejar el fin de cursos con sus compañeros. Quieren comprar dulces, pastel y refrescos, pero necesitan saber cuánto dinero deben pedir a sus compañeros.
5. Rosalí y Rocío están trabajando en equipo en el proyecto final de la materia “Ciencias Naturales”. Ellos no saben cómo planear y dividir el trabajo, de tal manera que puedan entregar el proyecto completo, a tiempo, y que todos los miembros del equipo trabajen. Piensa en un proyecto que hayas trabajado en equipo, ¿cómo repartieron el trabajo?, ¿el proyecto cumplió con lo que solicitó el profesor?, ¿lo terminaron a tiempo?, ¿crees que si se hubiera aplicado el Pensamiento computacional hubieras tenido mejores resultados?, ¿qué cambiarías?
6. Piensa en un problema en tu vida cotidiana, propón una solución utilizando el Pensamiento computacional. Identifica la información disponible para solucionar el problema (entrada); qué es lo que quieres obtener como resultado al resolver el problema (salida) e identifica el proceso que seguiste para solucionar el problema (relación entrada/salida).
7. Considera la dinámica de un partido de tenis (o algún otro deporte de competencia). Aplica el Pensamiento computacional para definir una estrategia que defina al ganador. ¿En qué consiste el problema? ¿Qué datos se necesitan para resolverlo?
8. Con frecuencia se considera que el Pensamiento computacional es aplicable solamente a problemas de ingeniería o matemáticas; sin embargo, puede aplicarse a prácticamente todas las áreas de la actividad humana. Sugiere problemas que podrían surgir en las áreas siguientes y cuya solución podría apoyarse en el Pensamiento computacional: derecho, música, economía, poesía, gastronomía.

9. Utiliza la metodología de George Polya para resolver los siguientes ejercicios. Para evaluar el análisis es importante que incluyas los siguientes pasos en cada situación planteada:
- Entiende el problema (descríbelo con tus propias palabras).
 - Sugiere una solución (haz un diagrama, emplea prueba y error, plantea una ecuación, reconoce patrones, etcétera).
 - Implementa la solución.
 - Mira hacia atrás y reflexiona si tu respuesta es correcta, si puedes hacerlo de otra manera, o si existen otras soluciones más viables.
10. Ordena los números del 1 al 5 en la siguiente gráfica de tal manera que la suma de los que están acomodados de forma horizontal sea igual a los que están de manera vertical.



11. La cuota de entrada para grupos grandes a un museo es de \$ 200.00 mn más \$ 45.00 mn por cada integrante. Si se quiere gastar un máximo de \$ 800.00 mn, ¿cuántas personas pueden ir?
12. Imagina que le compras a tu amigo un sello por 15 pesos, después se lo vendes por 20, enseguida se lo vuelves a comprar por 25 y al final se lo vendes por 30. ¿Cuál fue tu ganancia o pérdida? Tip: tú y tu amigo tienen cada uno cierta cantidad de dinero.

2.2 Datos y expresiones

Hoy en día vivimos inmersos en la llamada Era de la información. Los sistemas informáticos no cesan de producir cambios en la forma de trabajar de cada persona y de las organizaciones, debido a su capacidad de transformar **datos** en **información**. Estos sistemas basados en computadoras son de uso cotidiano para la creación, el almacenamiento y la transferencia de información. Además, gracias a la Web, millones de personas tienen acce-

so fácil e inmediato a una gran diversidad y cantidad de información. Pero seguro te preguntarás ¿qué es la información?

La **información** es un conjunto organizado de datos que han sido procesados y constituyen un mensaje que cambia el estado de conocimiento del sujeto o sistema que recibe dicho mensaje. Es la materia prima fundamental para el responsable de la toma de decisiones, porque aumenta su conocimiento. Basado en la información, el usuario puede tener una mayor certeza en el momento de desarrollar la solución adecuada a un determinado problema y tomar decisiones con mayor certidumbre entre un conjunto de alternativas posibles.

2.2.1 Los datos

La información es uno de los recursos más valiosos e importantes en cualquier actividad de nuestra vida diaria o en el quehacer profesional. Sin embargo, es común que este término se confunda con el de datos.

La palabra **dato** proviene del latín *datum* que significa “lo que se da” y el Diccionario de la Real Academia Española lo define como “información dispuesta de manera adecuada para su tratamiento por un ordenador” (2015). Para Norton (2008) son: “hechos o piezas individuales de información que por sí mismos no tienen mucho sentido para las personas”.

En Informática, **dato** es la expresión general que describe las características de las cosas sobre las cuales aplicamos un **proceso**. Se podría decir que es la mínima expresión susceptible a convertirse en información, de la misma manera que la **célula** es la unidad estructural y funcional de los seres vivos. Cuando se describe algo se hace mediante el manejo de datos, también llamados atributos.

Se dice que Juan es alto, delgado, de sexo masculino, que obtuvo la calificación de “Excelente” en el último trabajo de control y que es inteligente, se está describiendo a Juan mediante sus datos o atributos.

Veamos un ejemplo: ¿cuántos datos se desprenden de la expresión “La estatura de Juan es 1,73”? Un razonamiento sutil nos daría como respuesta tres datos: el nombre de la persona, la estatura de la persona y el sexo de la persona (tabla 2.3).

Tabla 2.3

Nombre	Estatura	Sexo
Juan	1.73	M

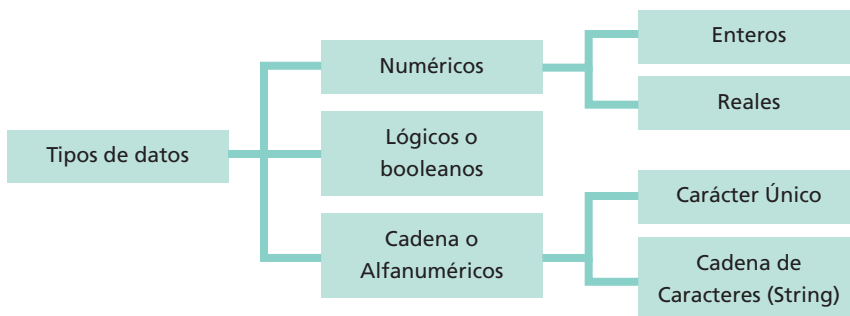
A un conjunto de datos existentes alrededor de una misma entidad se le denomina **registro** o **artículo**, y cada uno de los datos independientes de un registro se denomina **campo**.

Procesando estos datos, atributos o campos se puede llegar a un **conocimiento** más profundo acerca de Juan y esto nos puede llevar a una certera **toma de decisión** con respecto a Juan.

Por tanto, los datos son la representación de realidades concretas en su estado primario, de ahí que un dato pueda ser la palabra José, el número 69, la foto de tu carnet de identidad, el peso de un vehículo, pero ese valor o símbolo no tiene sentido por sí mismo, por el contrario, si es procesado adecuadamente, puede decirnos que ese 69 representa los kilos de José. Para representar esas realidades es posible usar varios tipos de datos como son: números, letras, imágenes, etcétera.

2.2.1.1 Clasificación de los datos

La utilización de los datos depende del lenguaje de programación que utilices para desarrollar tu aplicación y del tipo de dato adecuado para almacenar determinada característica. En la mayoría de los lenguajes de programación hay tipos de datos primitivos, a partir de los cuales se definen las variables que se deben utilizar. Los datos primitivos se clasifican usualmente en: numéricos, lógicos y de carácter (esquema 2.3).



Esquema 2.3 Clasificación de los datos.

Recuerda que el único alfabeto que comprenden los sistemas informáticos son los códigos binarios, o sea, 0 (ceros) y 1 (unos), de ahí que, increíblemente y por suerte hasta el momento, a pesar de los avances de la tecnología hoy en día los sistemas informáticos solo pueden procesar estos tres tipos de datos y el resto de los atributos de las cosas se reducen a estos mismos tipos: **datos numéricos**; **datos de cadenas o alfanuméricos**, **datos lógicos** o **de Boolean** (verdadero o falso). Inclusive los **datos audiovisuales** y **físicos** se reducen a estos tipos de datos. Algunos de estos se definen de la manera siguiente:

- **Númericos**: son números enteros o decimales, positivos o negativos, con los que se van a realizar operaciones aritméticas y/o lógicas. Algunos ejemplos de la vida cotidiana son: la estatura y peso de una persona, los coeficientes de una ecuación, los kilómetros recorridos por un automóvil, etcétera.
- **Alfanuméricos (texto)**: son aquellos que contienen cualquier combinación de letras, números y caracteres especiales con los que no se van a realizar operaciones aritméticas como: nombre, sexo, estado civil, dirección, ciudad, etcétera.
- **Audiovisuales**: son los que se pueden ver o escuchar; por ejemplo: voz, música, gráficas, dibujos, fotografías, secuencias de video, entre otros.
- **Físicos**: son aquellos datos que pueden ser capturados del medio ambiente: la intensidad de la luz de una habitación, la temperatura de un edificio, la presión arterial de una persona.

2.2.1.2 ¿Cómo se almacenan los datos en una computadora?

Cuando se introducen datos a la computadora o cuando se procesan, se realizan operaciones aritméticas y lógicas que generan resultados que deben ser almacenados dentro de la memoria RAM para ser utilizados posteriormente, ya que es la que se puede manipular a través del uso de variables y constantes.

Los conceptos de variable y constante guardan cierta relación con sus homólogos en matemática, pero no deben entenderse igual de manera literal como se verá a continuación.



Definición

Variable (en informática): es un espacio localizable de memoria que puede guardar un dato que puede cambiar.

Una **variable** posee un identificador que representa la dirección simbólica de una localidad de memoria en la que se puede guardar valores de forma temporal, ya que sus contenidos pueden cambiar, pero mantienen un mismo valor durante un período de tiempo.

Imagina que la memoria principal está dividida en pequeños segmentos que identificas con un nombre, como si fuera un casillero donde dejan las cartas. A cada uno le asignas un nombre para identificarlo y en ellos puedes guardar diferentes valores, pero solamente uno a la vez; es así como funciona, como un apartado postal que tiene un nombre que lo identifica de todos los demás y cuyo contenido está cambiando constantemente (figura 2.11).

DIR MEMORIA	VARIABLE	VALOR
0xFFFFFFFF0	num1	5
0xFFFFFFFF1	punt	
0xFFFFFFFF2		
0xFFFFFFFF3		
-		
-		
-		

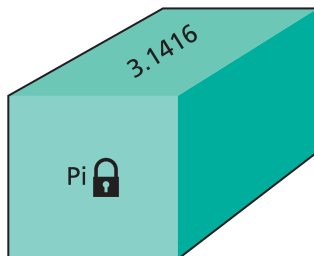
Fig. 2.11

Existen ciertos valores que no cambian durante el proceso de resolución de un problema, a estos se les conoce como **constantes**. Al igual que la variable, es un identificador que representa la dirección simbólica de una localidad de memoria en la que se guarda un valor que permanece en el tiempo (no puede cambiar). La figura 2.12 muestra cómo en una constante, a diferencia de una variable, su contenido está bloqueado y no puede cambiar mientras pasa el tiempo.



Definición

Constante (en informática): es un espacio localizable de memoria que guardar un dato que **No** se puede cambiar.



Constante

Fig. 2.12

2.2.1.3 Lineamientos para asignar nombre a las variables y a las constantes

Es importante tener en cuenta que cada una de las variables y constantes debe tener un nombre que las identifica de forma única. Dependiendo del lenguaje de programación que se emplee, las reglas de nomenclatura varían; sin embargo, cumplen con los siguientes lineamientos que se deben tomar en cuenta:

1. Solo contienen caracteres alfabéticos, números y guion bajo.
2. Comienzan con un carácter alfabético.
3. Son nemotécnicos, o sea, significativos; es decir, están relacionados con el valor que guardan. Por ejemplo, la variable “peso” indica que el contenido es el peso de algo o alguien.
4. No se extienden a más de 255 caracteres, aunque esto puede variar de acuerdo con el lenguaje de programación con que se trabaje.

Ejemplo de **identificadores de variables y constantes**:

- Contador - Correcto
- 4_Calificación – Incorrecto (comienza con número)
- Mi contador – Incorrecto (posee espacio)
- +Abc – Incorrecto (posee un símbolo inadecuado)
- Var - Correcto

Un ejemplo de **constante** es el número *Pi*, que es igual a **3,1416...**

Generalmente en la vida práctica una “gaveta” en la que se guardan zapatos tiene forma y posiciones diferentes que las gavetas en las que se guardan camisas, otro ejemplo muy evidente es que el estuche en que guardas un violín es muy diferente al estuche en que guardas una trompeta. Por eso, en la mayoría de los lenguajes de programación antes de guardar un dato resulta necesario inicialmente declarar el tipo de dato que se va a guardar, para que el sistema prepare o acondicione el espacio que va a reservar para ese dato.

2.2.1.4 Datos estructurados

Los **datos estructurados** son valores más complejos porque se derivan de los simples y contienen relaciones definidas entre ellos. Ejemplos de datos estructurados son los arreglos y las listas, árboles y grafos. Entre otros están:

- **Númericos:** es el tipo de dato que maneja los valores que pueden ser enteros o reales:
 - Enteros: son números sin punto decimal como el cero y toda la gama de positivos y negativos. Algunos ejemplos son: 50, 1000, -500, 8 500, -15 600.
 - Reales: almacenan números muy grandes con una parte decimal; pueden ser positivos o negativos, algunos ejemplos son: 5,5; - 10; -33,850; 139; 152 000; 25.
- **Alfanuméricos:** en este tipo de datos se encuentran los de carácter y los de cadena de caracteres:
 - Carácter: pueden contener solamente 1 carácter, ya sea un dígito (0, 1, 2,...9); una letra (a, b, c, ...z), o un símbolo especial (#, \$, ^, *, %, /, !, +, -, ...).
 - Cadena de caracteres: resultan de la combinación de caracteres alfabéticos, especiales y numéricos; deben estar delimitados por comillas """. Algunos ejemplos son: "María, N. L."; "7-2058357"; "Ave. Colón n.º 123"; "Juan Pérez".
- **Lógicos:** en esta clasificación se encuentran los booleanos. Estos datos solamente pueden tomar uno de dos valores: verdadero (**true**) o falso (**false**). Se utilizan para representar las alternativas, **sí** o **no**, a deter-

minadas condiciones. Por ejemplo, cuando se necesita definir si un determinado número es primo o no, la respuesta puede ser solamente verdadera o falsa.

2.2.2 Expresiones

Las operaciones que se realizan sobre las variables y/o constantes se denominan **expresiones** y están definidas por una serie de operadores que adquieren una *sintaxis diferente según el lenguaje de programación que se utilice*. No obstante, de manera genérica se va a proponer lo siguiente:

Operador de asignación (:=)

Si se desea que dentro de la variable **Volumen** se coloque el valor 5 se va a escribir:

Volumen: = 5

Existen diferentes tipos de operadores:

- Operadores aritméticos: la jerarquía de los operadores es similar a lo que aprendiste en matemática, primero se opera lo que este entre paréntesis, luego la potencia, y la multiplicación, la división, la división entera y el módulo según aparezcan, y finalmente la suma y la resta (tabla 2.4).

Tabla 2.4

Operador	Operación	Ejemplo	Resultado
<i>paréntesis</i>	Agrupamiento	$(2+3)*5$	25
$\wedge$	Potencia	4^4	16
*	Multiplicación	$5,25*3$	15,75
/	División	$100/4$	25
+	Suma	$2+4$	6
-	Resta	$4-2$	2
<i>Mod</i>	Módulo de la división entera	$4 \bmod 3$	1

Operador	Operación	Ejemplo	Resultado
<i>Div</i>	División entera	8 div 3	2
<i>Sqrt</i>	Raíz cuadrada	sqrt(16)	4
<i>Abs</i>	Valor absoluto	abs(-4)	4
<i>random</i>	Número aleatorio	random(10)	Un número al azar entre 1 y 10

- Operadores relacionales: son los operadores que sirven para comparar (tabla 2.5).

Tabla 2.5

Operador	Operación	Ejemplo	Resultado
=	Igual que	"Lalo"="Lola"	Falso
<>	Desigual a	"Lalo"<>"Lola"	Verdadero
<	Menor que	4 < 8	Verdadero
>	Mayor que	4 > 8	Falso
<=	Menor o igual	5 <= 12	Verdadero
>=	Mayor o igual	5 >= 12	Falso

- Operadores lógicos: son operadores que permiten conformar condicionales a partir de expresiones simples y pueden ser de tres tipos:
 - Conjunción (y)
 - Disyunción (o)
 - Negación (No)
- Operadores alfanuméricos: es un operador de Concatenación: Cad1 *Unir* Cad2 = Cad1Cad2. Por ejemplo

María *Unir* Elena = MaríaElena



Recuerda que...

Los paréntesis permiten alterar el orden lógico de las operaciones.

Ejemplo: $A/(2+3)$

Según el orden de operaciones previsto se realiza siempre la división antes que la suma, sin embargo, al estar la suma entre paréntesis debe realizarse la suma antes y luego la división.

2.2.3 Contadores y acumuladores

Como se planteó anteriormente una variable es una localización de memoria en la que se almacena un dato. Pero entonces surge la pregunta, ¿qué existe en esa localización antes de que en ella se deposite un dato?, como mismo ocurriría con una gaveta, se podría decir que esta está vacía o **podría tener otro dato que fuese colocado anteriormente**. Esta es la causa por la cual al trabajar con variables es necesario una operación que se denomina **inicialización**.



Definición

Inicialización de una variable: inicializar una variable es asignarle de manera explícita un valor inicial al espacio de memoria que la variable ocupará. Esta operación es imprescindible si la variable está sujeta a cambios como consecuencia de operar con su contenido.

Analicemos que podrían significar las siguientes expresiones:

- $A = 2$ (a un espacio de memoria al que se le ha llamado A, se le asigna o deposita el valor 2).
- $Var = "A"$ (a un espacio de memoria al que se le ha llamado Var, se le asigna o deposita el valor de cadena A).
- $X = X + 1$ (como debes estar pensando, esta expresión desde el punto de vista matemático es absurda, ya que bajo el supuesto de que X es un número entero, la expresión plantea que el número X es igual a su sucesor $X+1$, y como sabemos esto nos llevaría al absurdo de que $0 = 1$, algo así como que el todo es igual a la nada).

Sin embargo, el hecho de que la Informática tenga tanta relación con la matemática hizo que un mismo signo $=$ se aplicara en ambas ciencias de manera distinta. En matemática el signo $=$, significa **igualdad**, mientras que en informática el signo $=$ significa **asignación**. Igualdad pretende decir, que lo que está a la izquierda es igual a lo que está a la derecha y asignación significa que lo que está a la derecha deberá reemplazar el valor que está a la izquierda, por tal motivo la expresión informática $X = X + 1$ quiere decir que donde estaba X se cambie por el valor que estaba, o sea, el valor de X , adicionándole 1. ¿Interesante verdad?

Tan delicado es el asunto que algunos lenguajes de programación han abolido el **signo de igualdad** como equivalente del **signo de asignación**.

Al escribir correctamente la expresión $X=X+1$, se escribirá $X:= X+1$. Esta expresión representa un **contador**.



Definición

Contador: un contador en Informática se entiende como una expresión del tipo: $C:=C+1$

Para que te sea más fácil entender, los contadores son variables que cuentan el número de eventos ejecutados dentro de un algoritmo. Este aumenta o disminuye su contenido en un valor constante. El contador inicia generalmente en 0 o 1. Por ejemplo, en el videojuego Mario Bros se utiliza un contador para almacenar las vidas, el tiempo o la cantidad de monedas.

Las operaciones básicas que se llevan a cabo con una variable contador son:

1. Inicialización: **contador** = **valor inicial**. Esta operación asigna el valor inicial de la variable contador. Por ejemplo: **vidas** = 1.
2. Incremento o decremento: para el incremento: **contador** = **contador** + **constante**, y para el decremento: **contador** = **contador** - **constante**.

Dichas operaciones se realizan cada vez que ocurre el evento que se desea contar. Usualmente el incremento o decremento corresponde a la constante, por ejemplo: **vidas** = **vidas** + 1. Desde luego, pueden utilizarse otras constantes, por ejemplo: **total** = **total** + 5.



Definición

Acumulador: un acumulador o sumador en Informática se entiende como una expresión del tipo: $S := S + k$. Donde k es valor arbitrario.

Los acumuladores incrementan o decrementan su contenido en un valor variable. La acumulación inicia generalmente en 0 o 1. La operación básica que se debe realizar con ellos es:

1. Inicialización: **acumulador = valor inicial**. Esta operación asigna el valor inicial de la variable acumulador y depende del tipo de operación que se realizará, por ejemplo: **energía = 100**

2. Incremento o decremento. Para el incremento:

acumulador = acumulador + variable,

y para el decremento:

acumulador = acumulador - variable.

Dichas operaciones se realizan cada vez que ocurre el evento que se desea acumular. Por ejemplo: **vitalidad = vitalidad - impacto.**

2.2.4 Descripción de algoritmos

El vocablo **algoritmo** es de origen árabe y proviene del sobrenombre del matemático Al-Khowarizmi. Muchas personas cuando escuchan la palabra algoritmo, piensan que es demasiado sofisticada y que está exclusivamente relacionada con las ciencias. Sin embargo, los algoritmos están ampliamente relacionados con nuestra vida cotidiana.

En grados anteriores viste que un algoritmo **es una serie de pasos ordenados que se siguen para resolver un problema**. No obstante, es importante que sepas que un algoritmo debe siempre terminar luego de un número finito de pasos, pues de lo contrario no se puede garantizar alcanzar una solución. Es necesario precisar que aun cuando no es posible arribar a una solución, los algoritmos deben tener un paso final o punto de salida. Por último, cada paso del algoritmo debe ser bien definido, lo que significa que debe ser especificado de manera precisa, rigurosa y sin ambigüedades para que se entienda exactamente el significado de estos y los resultados intermedios queden definidos de manera única y solo de-

pendan de los datos de entrada y de los resultados de los pasos anteriores. Es decir, que la misma secuencia de pasos con los mismos datos de entrada, no puede dar lugar a más de un resultado.

En la vida cotidiana los algoritmos se emplean ampliamente, por ejemplo, en las recetas para preparar tu desayuno, en la instalación de una aplicación en tu Smartphone (teléfono inteligente) o en preparar la receta para hacer un pastel de queso. Como sabes, este pastel tiene diferentes recetas para prepararse. Por esta razón, una persona que está tratando de hacerlo tiene varias maneras que puede utilizar para lograr su objetivo. Los resultados no serán los mismos porque algunas recetas requieren mayor tiempo de preparación, tienen ingredientes costosos o difíciles de conseguir, tienen demasiada azúcar, u otras características que afectan el resultado del producto final.

Al igual que en las recetas, un problema en particular puede resolverse de varias maneras. Por esta razón, hay varios algoritmos que se pueden solucionar con diferentes ventajas y desventajas, algunos tardan más tiempo en encontrar una solución u otros consumen más memoria.

2.2.4.1 Construcción de algoritmos

La construcción de algoritmos se va corresponder con las siguientes ideas:

1. **Secuenciación:** es la idea que plantea que las acciones de los algoritmos se irán realizando de arriba hacia abajo, comenzando por la primera acción hasta la última.
2. Las acciones pueden ser diferentes, pero siempre serán de lectura, procesamiento y salida de información.
3. Es conveniente comenzar un algoritmo con un identificador de lo que hace, así como finalizarlo con la palabra **fin**.



Recuerda que...

Los **algoritmos** pueden ser representados de cuatro maneras diferentes, mediante:

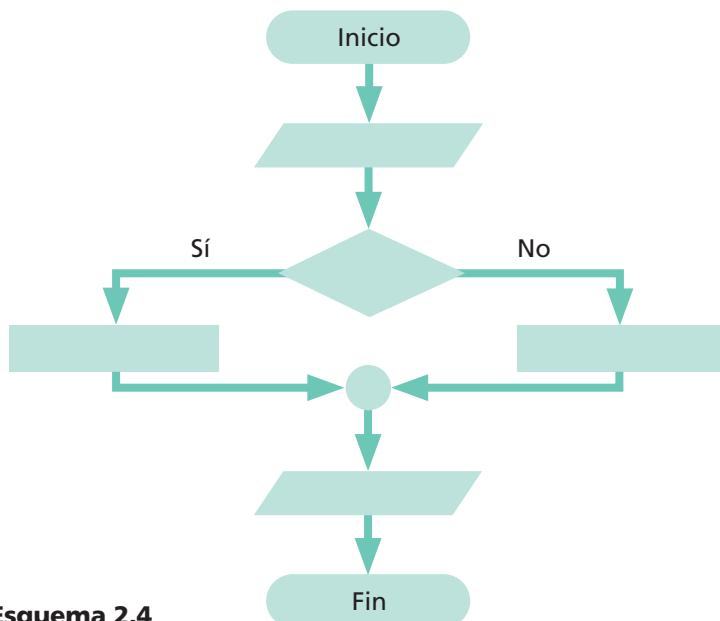
- El lenguaje natural (español)
- Un diagrama de flujo
- Un pseudocódigo
- Un lenguaje de programación o implementación

Como debes haber recordado los algoritmos se representan de varias formas, pero las descripciones en lenguaje natural pueden ser ambiguas y extensas, mientras que los diagramas de flujo y el pseudocódigo evitan las ambigüedades del lenguaje natural al ser representaciones más estructuradas de los algoritmos; además, son independientes de los lenguajes de programación. Por estas razones, se van a analizar los diagramas de flujo y el pseudocódigo.

Expresión de algoritmos mediante diagramas de flujo y pseudocódigo

Antes de iniciar el análisis y la construcción de algoritmos mediante diagramas de flujo, es válido señalar que estos no son solo importantes en Informática, sino en todos los procesos que llevan consigo una secuencia lógica. Recuerdas cuándo estudiaste este tipo de esquema en octavo grado a través de los organizadores gráficos, de igual forma el diagrama de flujo se utiliza para representar esa secuencia ordenada de acciones que permiten solucionar un problema.

Los **diagramas de flujo** son la representación gráfica y visual de cada paso del algoritmo por medio de símbolos sencillos de fácil comprensión que representan el flujo de los datos en la solución de un problema, y las operaciones ejecutadas sobre los datos (esquema 2.4).



Esquema 2.4

Hay símbolos aceptados como estándar, a partir de las propuestas de organizaciones como: American National Standards Institute (ANSI) y la International Organization for Standardization (ISO) (figura 2.13).

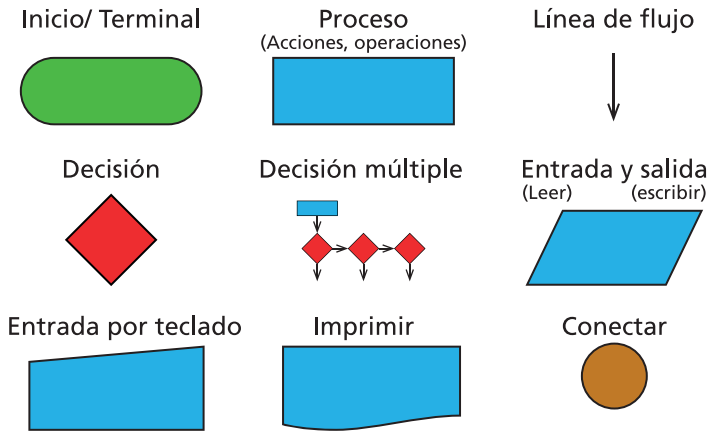


Fig. 2.13 Símbolos estándar para los diagramas de flujos.

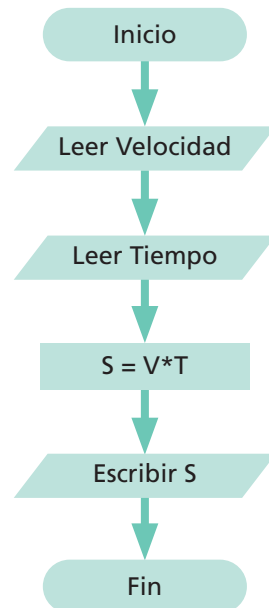
Ahora se propone analizar estos tres ejemplos clásicos de representación de un algoritmo utilizando el diagrama de flujo.

Ejemplo 1

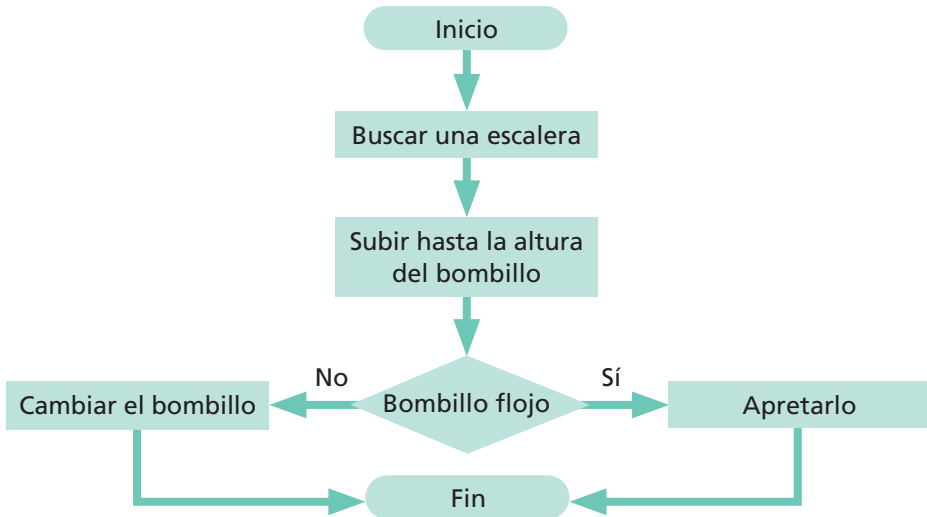
Describir un algoritmo que permita calcular el espacio recorrido por un cuerpo que se mueve con Movimiento Rectilíneo Uniforme (MRU) si se conocen la velocidad, el tiempo transcurrido en el desplazamiento y la fuerza de rozamiento (esquema 2.5).

Ejemplo 2

Al interactuar con el interruptor de la luz del comedor se constata que la luz no enciende. Elabore un diagrama de flujo para resolver este problema (esquema 2.6).



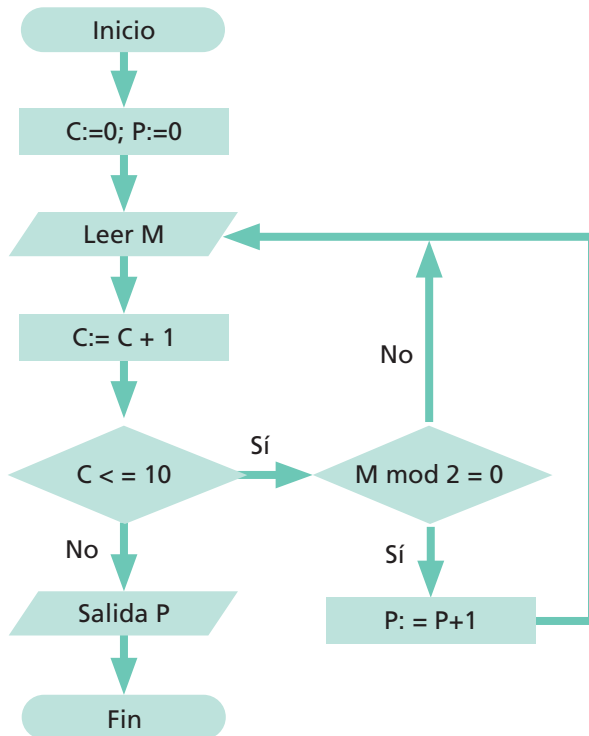
Esquema 2.5



Esquema 2.6

Ejemplo 3

Dado un conjunto de 10 números enteros, determinar cuántos son pares (esquema 2.7).



Esquema 2.7

Posterior a la elaboración de los tres ejemplos, es interesante realizar el siguiente análisis:

1. Los tres diagramas de flujo poseen inicio y fin.
2. En el ejemplo 1 las acciones ocurrirán invariablemente siempre de la misma manera, o sea, en la misma secuencia.
3. En los ejemplos 2 y 3 está presente la figura que representa la existencia de condicionales, o sea, las acciones no siempre ocurrirán en la misma secuencia. Los datos una vez correrán por la rama de los **Sí** y otras veces podrían correr por la rama de los **No**.
4. Para lograr un resultado en los ejemplos 1 y 2, las acciones ocurrirán una sola vez.
5. Para lograr un resultado en el ejemplo 3, las acciones ocurrirán repetidamente (en particular 10 veces).

Los ejemplos realizados son representantes genuinos de una de las más importantes ideas de las ciencias de la computación. Por ello al resolver un problema con un sistema informático el flujo de los datos tendrá características **secuenciales** (algoritmo secuencial), características de **flujo condicionado** (algoritmos alternativos) o características **repetitivas** (algoritmos cíclicos o iterativos). Se propone a continuación el estudio del teorema fundamental de la programación estructurada.



Teorema

Teorema fundamental de la programación estructurada: establece que toda función computable puede ser implementada en un lenguaje de programación que combine solo tres estructuras lógicas o de control de flujo. Esas tres formas son:

- Secuencial: ejecución de una instrucción tras otra.
- Alternativa: ejecución de una de dos instrucciones (o conjuntos), según el valor de una variable booleana.
- Cíclica: ejecución de una instrucción (o conjunto) mientras una variable booleana sea 'verdadera'. Esta estructura lógica también se conoce como **ciclo** o **bucle**.

Expresión de algoritmos mediante pseudocódigo

Los lenguajes naturales como el español o el inglés son extremadamente ricos a tal extremo que gracias a estas características, tanto la prosa como el verso devienen del talento de oradores y escritores, cada uno

empleando recursos lingüísticos como resultados de su imaginación, creatividad y talento, y en esto influye indiscutiblemente la versatilidad y el carácter polisemántico de los lenguajes naturales. Al leer una obra literaria, por ejemplo, imaginamos lo escrito por el autor, pero desde una posición personalizada, o sea, acorde con nuestras características individuales que evidentemente dependerán de múltiples factores (cultura, idiosincrasia, contexto, etc.). Por lo antes dicho, la descripción de un algoritmo mediante el lenguaje natural puede verse afectado por los elementos antes mencionados que le transfieren a los algoritmos ciertos elementos de ambigüedad. Para evitar este problema se ha creado el **pseudocódigo**.



Definición

Pseudocódigo: es la formalización o elección de un subconjunto de palabras del lenguaje natural mezclado con algunos símbolos para expresar algoritmos.

El **pseudocódigo** no llega a ser un lenguaje de programación y por tanto, no es entendible por sistemas informáticos, pero contiene muchas características comunes a todos los lenguajes de programación, de ahí su valor didáctico y su importancia para aprender a programar computadoras.

Ahora corresponde el análisis de los tres ejemplos clásicos de representación de un algoritmo, utilizando el pseudocódigo.

Ejemplo 1

Describir un algoritmo que permita calcular el espacio recorrido por un cuerpo que se mueve con MRU si se conocen la velocidad y el tiempo transcurrido en el desplazamiento.



Definición

Estructuras de control secuencial: son aquellas en que las acciones se suceden una detrás de la otra, sin que ninguna condición altere el sentido de ejecución de las acciones.

Se pone de manifiesto en el ejemplo número 1.

Inicio

Leer V

Leer t

*S := V * t*

Escribir S

Fin

Ejemplo 2

Al interactuar con el interruptor de la luz del comedor se constata que la luz no enciende. Elabore un algoritmo en pseudocódigo para resolver este problema.



Definición

Estructuras de control alternativa: son aquellas en que las acciones se suceden una vez, pero en dependencia del cumplimiento o no de ciertas condiciones.

Se pone de manifiesto en el ejemplo número 2.

Inicio

Buscar escalera

Subir hasta la altura del bombillo

Sí (Bombillo flojo) entonces

Apretarlo

sino

Cambiarlo

Fin si

Fin

Ejemplo 3

Dado un conjunto de 10 números enteros, determinar cuántos son pares.



Definición

Estructuras de control cíclica: se conocen también como iterativas. Son aquellas estructuras en que las acciones se repiten más de una vez, un número determinado de veces o hasta tanto se cumpla una condición.

Se pone de manifiesto en el ejemplo número 3.

Inicio

C:=0

P:=0

Mientras C < 10

Leer un número M

C:= C+1

Si (M mod 2) = 0 entonces

P:= P + 1

Fin si

Fin Mientras

Escribe "Cantidad de números pares:" Unir P

Fin



Algo de historia

Al-Juarismi: fue un matemático, astrónomo y geógrafo persa musulmán, que vivió aproximadamente entre los años 780 y 850, al que se le atribuye el origen de las palabras "álgebra" y "algoritmo". Es considerado como el padre del álgebra y como el introductor de nuestro sistema de numeración denominado arábigo.

2.2.5 Elementos de programación

¿Haz programado alguna vez?, la pregunta te podrá parecer un tanto extraña, sin embargo, es importante que sepas que sí, que tú programas y de manera permanente. Imagínate situaciones como estas:

- Ejemplo 1: todos los días de la semana, en las mañanas, excepto sábados y domingos, sueles hacer lo siguiente: "Me levanto, me cepillo los dientes, desayuno y me voy para la escuela".
- Ejemplo 2: te levantas un domingo con la idea de ir a la playa y piensas: "si me asomo a la ventana y está lloviendo, entonces me quedo en casa a estudiar, sino, tomaré el autobús y me encontraré con mis amigos para irnos a la playa".

Como ves, situaciones como las aquí enunciadas podrías enumerar cientos, día tras día, pues estos ejemplos están muy cercanos a lo que se denomina **algoritmo** y los algoritmos constituyen el concepto básico de

la programación de los sistemas informáticos. En la vida cotidiana, se emplean para resolver problemas. Algunos ejemplos pueden ser:

- a. Las indicaciones que aparecen en manuales de aparatos o dispositivos, las que usamos generalmente al comprarlos.
- b. Algunos ejemplos en Matemática son: el **algoritmo de multiplicación**, para **calcular** el producto; el **algoritmo de la división** para calcular el cociente de dos números; el **algoritmo de Euclides** para obtener el máximo común divisor de dos enteros positivos; el **método de Gauss** para resolver un sistema lineal de ecuaciones o el procedimiento que se emplea para saber si un número es primo.
- c. Balanceo de una fórmula química.
- d. El cálculo de la trayectoria de un proyectil con movimiento uniformemente variado.
- e. La determinación de la acentuación gráfica de palabras agudas, llanas y esdrújulas.
- f. La identificación del período histórico a que pertenece un hecho a partir de la fecha en que ocurre.

2.2.5.1 Lenguajes de programación

En séptimo grado, al estudiar el concepto de **software** entendimos que los **software** se dividen en tres categorías:

- a. **Software** básico (sistema operativo, manipuladores, BIOS, etcétera).
- b. Sistemas de aplicaciones (procesadores de texto, graficadores, hojas de cálculo, sistemas de gestión de bases de datos, videojuegos, **software** educativo...).
- c. Lenguajes de programación.

En séptimo y octavo grado se estudiaron los **software** del tipo a y b, corresponde entonces estudiar los **software** de tipo c, o sea, los lenguajes de programación, que son las formas en que se escriben los lenguajes informáticos.



Definición

Programa informático: es un algoritmo que se escribe de manera tal que resulte comprensible para un sistema informático (una computadora, un teléfono inteligente, una tableta, etcétera).

Tipos de lenguajes de programación

Los programas se escriben en lo que se denomina **código fuente**, que son las instrucciones que el programador indica que deben ser ejecutadas a fin de realizar las acciones para las que el programa ha sido diseñado. Sin embargo, este código no lo puede comprender directamente el ordenador y debe ser traducido al único lenguaje que el ordenador puede interpretar: el **lenguaje máquina** o **código binario**. Para realizar esta transformación de código se utilizan unos traductores llamados **compiladores** e **intérpretes**, que convierten las instrucciones dadas por el programador en instrucciones comprensibles por un ordenador, es decir en el llamado **código objeto**, que es el archivo traducido a lenguaje máquina. Los **tipos de lenguajes de programación** son los siguientes:

- Lenguaje de máquina o lenguaje binario: es el lenguaje de programación que comprende de manera directa los sistemas informáticos. Está compuesto por un alfabeto de dos símbolos: 0 y 1 (ceros y unos), formando cadenas binarias que conforman las instrucciones que la CPU (Unidad Central de Procesamiento) realiza.
Como es de suponer, resulta extremadamente difícil programar en este lenguaje propio solo de máquinas, como su nombre indica; por tal motivo han surgido otros lenguajes más fáciles de comprender para el hombre.
- Lenguajes de bajo nivel o Lenguaje ensamblador: fue el primer intento de facilitar la escritura de programas acercándola un poco a la manera de escribir las instrucciones con abreviaturas del lenguaje natural, como son ADD, MOV, XCHG, LD, etc. Algo significativo del lenguaje ensamblador es que es particular para cada microprocesador, o sea, es dependiente del dispositivo con que se trabaje. De cualquier manera, lo que se escribe en **ensamblador** deberá ser traducido al lenguaje de máquina.
- Lenguajes de alto nivel: son los más parecidos al lenguaje natural, en especial al idioma inglés, y son independientes del dispositivo, o sea, sirven para programar cualquier sistema informático solo en dependencia del sistema operativo con que se trabaje. Algunos son multiplataforma, es decir, funcionan con cualquier sistema operativo.



Saber más

Desde el punto de vista histórico los lenguajes de programación se pueden dividir en:

- Lenguajes de **1.ª generación** (1GL): (antes de 1950)
Se corresponden con los ya mencionados lenguajes de máquina.
- Lenguajes de **2.ª generación** (2GL): (1950-1955) Introducen los lenguajes ensambladores.
- Lenguajes de **3.ª generación** (3GL): (1956-1965) Poco a poco los intérpretes admitieron palabras más complejas y comenzaron a parecerse a los lenguajes naturales como el inglés, español, etc. En los años 70 se crean **PASCAL, ADA y PROLOG**. Mención especial debe atribuirse al lenguaje C, desarrollado por Dennis Ritchie, creador también del sistema operativo **UNIX**, que dio origen a **LINUX y Android**.
- Lenguajes de **4.ª generación** (4GL): (a partir de 1980) Se vinculan con la llamada **programación orientada a objeto** (POO). Se alcanza cada vez más la cercanía con el lenguaje natural y se crean entornos de desarrollo llamados IDE con interfaces gráficas de usuario (GUI). Representantes de esta generación son: **Java, C++, Delphi, Perl, PHP, Python, Visual Basic, Smalltalk, Ruby, Eiffel, Visual FoxPro**, etcétera.
En esta categoría aparece una familia de Lenguajes denominada **HyperTalk**, diseñadas en el mundo de las Macintosh y que son impresionantemente cercanas al lenguaje natural, al límite de poseer una gramática similar a los lenguajes naturales con fenómenos de sinonimia, pronombres personales, preposiciones, etc. Ejemplos de esta familia son **OpenScript, Transcript, Livecode Scripting**, entre otros.
- Lenguajes de **5.ª generación**: son lenguajes especializados en técnicas de inteligencia artificial.



Como se ha explicado, en realidad el único lenguaje que verdaderamente comprenden los sistemas informáticos es el lenguaje binario, o sea, ese que solamente está formado por 0 y 1, por lo tanto, en cualquiera de los casos se culmina realizando una traducción hacia 0 y 1 (ceros y unos).

Tipos de lenguajes según la forma de traducción que emplean

Como ya se ha dicho, el único lenguaje que comprenden los sistemas informáticos de manera directa es el llamado *lenguaje de máquina*. Esto significa que cualquier otra variante tiene que ser traducida a este lenguaje; pues bien, en dependencia de la manera en que se realiza la traducción, los lenguajes pueden ser *interpretados* (intérpretes) o *compilados* (compiladores).



Definición

Intérpretes: los lenguajes intérpretes realizan la traducción de manera simultánea, o sea, línea a línea del documento que constituye el programa. Traducen una línea e inmediatamente la ejecutan.

La principal ventaja de estos lenguajes radica en la facilidad en que se detecta un error de escritura o error sintáctico, ya que la línea que contiene un error, al intentar ejecutarse, pone de manifiesto el error inmediatamente y de esta forma se facilita la localización del error.



Definición

Compiladores: por el contrario de los lenguajes intérpretes, los compiladores traducen todas las líneas del programa antes de ejecutar ninguna de sus líneas, produciendo un documento que se denomina **código objeto**, que es el que queda listo para ser ejecutado por la máquina.

La ejecución de un programa compilado resulta siempre ser más rápido que desde un programa interpretado; sin embargo, la depuración o localización de errores sintácticos en la programación resulta ser más compleja.

Como has podido comprobar, la programación ha evolucionado y se ha adaptado al avance de los ordenadores y sistemas. Con el tiempo se han establecido diferentes criterios y técnicas de programación, no excluyentes entre sí, por ejemplo:

- Programación modular: consta de varias secciones o módulos que interactúan a través de llamadas a procedimientos. El módulo principal, encargado de llamar al resto de los módulos, integra el programa en su totalidad.
- Programación estructurada: está compuesta por un conjunto de técnicas evolucionadas que aumentan la productividad del programa. Se

basa en un proceso lineal y sencillo que se apoya en estructuras secuenciales, selectivas y repetitivas.

- Programación orientada a objetos: usa objetos y sus interacciones para diseñar aplicaciones. Los objetos son entidades que poseen una serie de propiedades destinadas a que los programas sean más fáciles de escribir, mantener y reutilizar.



Algo de historia

ADA: es un **lenguaje de programación orientado a objetos**, diseñado bajo encargo del Departamento de Defensa de los Estados Unidos. El nombre se eligió en conmemoración de lady Ada Augusta Byron (1815-1852), condesa de Lovelace, hija del poeta Lord George Byron, a quien se considera la primera programadora de la historia por su colaboración y relación con Charles Babbage, creador de la **máquina analítica**.

Comprueba lo aprendido

1. ¿Qué se entiende por programa informático?
2. ¿Qué lugar ocupa el concepto de *lenguaje de programación* en el concepto de *software*?
3. ¿Qué tipos de lenguajes de programación se discuten en este libro?
4. ¿Cuál es el único lenguaje que comprende un sistema informático de manera directa?
5. ¿Cómo se clasifican los lenguajes de programación según la forma en que producen su traducción al lenguaje de máquina?
6. ¿Por qué se crearon los lenguajes de alto nivel?
7. Investiga datos biográficos sobre la primera mujer programadora de la historia.
8. De qué lugar es el matemático al que se le atribuye el concepto de *algoritmo*. ¿Cómo se conoce este país hoy en día?

9. Elabora una línea de tiempo que recoja la evolución histórica de los lenguajes de programación.
10. ¿Qué se entiende por algoritmo?
11. Identifica procedimientos relacionados con la vida cotidiana o procedentes de otras asignaturas que puedan ser considerados algoritmos.
12. Escribe con tus palabras algoritmos para:
 - a) Cruzar una avenida.
 - b) Saber si un número dado es par.
 - c) Freír un huevo.
 - d) Ir de la casa a la escuela.
 - e) Pasar una aplicación de un teléfono a otro utilizando Zappia.
13. Identifica y comenta cuatro formas de expresar algoritmos.
14. Por qué el lenguaje natural no es un recurso idóneo para expresar algoritmos.
15. Realiza un cuadro comparativo de las técnicas de representación de algoritmos donde abordes las ventajas y desventajas de cada una.

2.3 La programación orientada a objetos y dirigida por eventos

Como ya sabes, existen básicamente cuatro maneras de expresar algoritmos, mediante:

- el lenguaje natural,
- el de diagramas de flujo,
- el de pseudocódigo,
- un lenguaje de programación.

En esta unidad aprenderás a **implementar algoritmos**, es decir, a describirlos mediante un **lenguaje de programación**. Hay cientos de lenguajes de programación y en principios la elección de uno en particular obedece a múltiples criterios, desde los que se ponen de moda hasta los que resul-

tan demasiado difíciles de aprender por poseer una sintaxis relativamente abstracta y críptica. Lo que sí es cierto es que todos los seres humanos estamos capacitados para leer y escribir en una lengua natural, por tal motivo, en la medida que el lenguaje de programación este cercano a un lenguaje natural, nos deberá resultar más asequible.

Los lenguajes orientados a objetos son la evolución lógica de la programación estructurada. La **programación orientada a objetos** (POO u OOP, según sus siglas en inglés) se basa en dividir los programas en pequeñas unidades lógicas de código. A estas pequeñas unidades lógicas de código se las llama objetos. Los objetos son unidades independientes que se comunican entre ellos mediante **mensajes**, los cuales poseen una serie de características especiales y pueden ser reutilizados.

Los objetos

La POO intenta describir de forma abstracta la forma de pensar de los seres humanos. Pensar en términos de objetos, es muy parecido a cómo lo haríamos en la vida real; por ejemplo, se diría que un auto es un objeto que tiene una serie de características (como podrían ser la marca, el modelo y el color) y una serie de funcionalidades asociadas (como ponerse en marcha o frenar). En el esquema de la POO, el coche sería el **objeto**; las **propiedades o atributos** serían las **características** (color, modelo, etc.); y los **métodos** serían las **funcionalidades asociadas** (ponerse en marcha, frenar, etc.). Un objeto de **software** mantiene sus características en una o más **variables** e implementa su comportamiento con métodos. Un método es una **función** o **subrutina** (pequeño programa) asociada a un objeto.

Las clases

En el mundo real, normalmente se tienen muchos objetos del mismo tipo; por ejemplo, hay muchas marcas de autos. Si se habla en términos de la POO, se puede decir que nuestro objeto auto es una instancia de una clase conocida como automóvil. Todos los autos tienen unas características (marca, modelo, color, motor, ventanas, ruedas, etc.) y unos comportamientos (ponerse en marcha, frenar, acelerar, parar, ir marcha atrás, etc.). Cuando los fabricantes de automóviles fabrican coches, aprovechan el hecho de que todos los autos deben compartir sus características comunes y elaboran modelos o plantillas que luego aplican a todos los autos de ese mismo

modelo. Esos modelos o plantillas son lo que se conocen como **clases**, y a cada auto que se fabrica utilizando esos modelos se les denomina **objeto**.

La herencia

Imagina ahora que el fabricante de automóviles, partiendo del modelo original, realiza dos modelos de auto distintos y a cada uno de ellos le otorga una característica diferente, realizando un modelo para autos con cambio manual y otro para autos con cambio de marchas automáticas. Cada modelo de automóvil hereda de su predecesor todos los atributos que este tenía (marca, modelo, color...) y sus propiedades (ponerse en marcha, frenar, acelerar...), y además cada uno incorpora un nuevo atributo (cambio). El primer modelo (clase automóvil) se convertirá en una **superclase**, ya que de ella se han derivado otras, y las clases generadas recibirán el nombre de **subclases**.

Envío de mensajes

Los objetos por sí mismos no tienen mucha utilidad; por ejemplo, un auto, sin una persona que lo conduzca, pierde gran parte de su significado. Para relacionar distintos objetos se emplean los **mensajes**, de esta manera, una persona consigue que un coche gire a la derecha moviendo el volante hacia ese lado y también se utilizan **métodos**, con los cuales se pueden incluir ciertos **parámetros** o **variables**; por ejemplo, al objeto auto se le podría enviar un mensaje con el método **girar**, indicándole como parámetro la dirección (derecha o izquierda). También se puede invocar a los métodos sin necesidad de pasar ningún parámetro, como se haría en el caso de los métodos **frenar** y **acelerar**. En términos generales, se podría decir que los **mensajes** son la forma de realizar acciones sobre los objetos.

Después de este acercamiento necesario a algunos conceptos importantes de la programación orientada a objetos y acordes con los principios curriculares del perfeccionamiento, cada centro podrá elegir las herramientas informáticas con que trabajar, siempre y cuando cumpla con los objetivos plasmados en el programa y eso significa que el profesor podrá emplear el lenguaje de programación que más le agrade; no obstante, existen razones que nos instan a proponer una herramienta muy moderna, poderosa y profesional que servirá de base a ulteriores desempeños en esta disciplina, sobre todo si se tiene en cuenta el estado del arte actual de

la Informática vinculada con la irrupción de los dispositivos móviles. Esta herramienta de programación se denomina **LiveCode**.

2.3.1 Breve reseña histórica de LiveCode, algunas de sus características



LiveCode se inspira en un lenguaje de programación mucho más antiguo denominado **HyperTalk**. Este lenguaje fue creado en 1987 por un programador estadounidense llamado Dan Winkler. Por lo visto, **Apple** se interesó mucho en este lenguaje y lo compró para incorporarlo en un producto suyo denominado **HyperCard**.

HyperTalk era un lenguaje de programación extraordinariamente sencillo (prácticamente era como hablar en inglés). Steve Wozniak (el cofundador) de Apple llegó a decir que HyperCard era el mejor software jamás escrito. A finales de los años 80 existían cientos de empresas trabajando en el ecosistema de HyperCard, incluso la todopoderosa **Oracle** ofrecía un **software** basado en HyperCard para conectarse a bases de datos de Oracle: **Oracle Media Objects** (la cual, luego derivó en **Oracle Forms**).

Hubo fabricantes que crearon un montón de "**plugins**" para HyperCard, primitivos sistemas de acceso a Internet, controladores de los puertos hardware del Macintosh para manipular dispositivos electrónicos, sistemas de animación y multimedia, etc. Actualmente, lenguajes de programación como **Apple Script** están basados en HyperCard.

LiveCode está inspirado completamente en el lenguaje HyperTalk (versión MetaCard) y la primera versión del lenguaje surgió en el año 2001. En aquella época se llamó **Revolution**, nombre que fue cambiado a **LiveCode** en el año 2010. Fue diseñado para aprovechar repositorios que le permitieran a la gente aprender a programar más rápido; es decir, la sintaxis y el vocabulario utilizado en LiveCode es similar a un lenguaje natural. La empresa que creó y desarrolla actualmente este lenguaje es **Runtime Revolution Ltd**, empresa escocesa con sede en Edimburgo.

LiveCode ganó el premio de la revista **Macworld (Macworld Annual Editor's Choice Award)** por "mejor desarrollo de **software** del año 2004" y a partir del año 2013 existen dos versiones de este lenguaje: una versión

propietaria y una completamente libre, la cual es gratis pero pone como condición que si se comercializa un programa de esta versión, es obligatorio publicar el código de forma pública (Licencia GPL).

Algunas *características de LiveCode* son las siguientes:

- Es **software** libre.
- Utiliza una metáfora muy intuitiva.
- Ha desarrollado multiplataformas (Windows, Linux, MacOSX, Android, IOs, UNIX y HTML 5). Su consigna es "Write once, run any where" (Escribe una vez, ejecútalo donde quieras...).
- Su lenguaje de programación es impresionantemente parecido al lenguaje natural (inglés).
- Es visual, basado en objetos y dirigido por eventos.
- Es muy versátil ya que es de propósito general, o sea, se pueden desarrollar disímiles aplicaciones para diferentes campos, incluyendo aplicaciones para móviles.
- Sobre él existe una amplia experiencia pedagógica.

¿Cómo es el lenguaje? LiveCode es un ambiente de programación basado en objetos y dirigido por eventos, con un lenguaje de programación muy sencillo, es casi como hablar en inglés; por ejemplo: Lo siguiente es una instrucción en LiveCode:

Put José into field Nombre of card Datos of stack Base

La traducción al español sería: ***Pon "José" dentro del campo Nombre de la Tarjeta Datos que pertenece al módulo o pila Base.***

LiveCode tiene aproximadamente 2600 (un poco más) sentencias de programación y estas se pueden extender mediante "*plugins*" o programándolas directamente en *lenguaje "C"*.

La filosofía o metáfora de la herramienta de desarrollo se basa en una "pila de tarjetas" o una "baraja de cartas", cada una de las cuales constituye **una ventana** de la aplicación. Existe una **pila principal** denominada "Main Stack" que contiene **Cards** o "tarjetas" (figura 2.14).

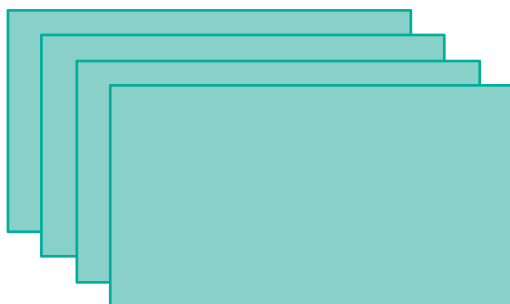


Fig. 2.14 Pila principal o Main Stack.

La **pila principal** puede a su vez estar acompañada por otras pilas, cada una con su sistema de tarjetas (figura 2.15).

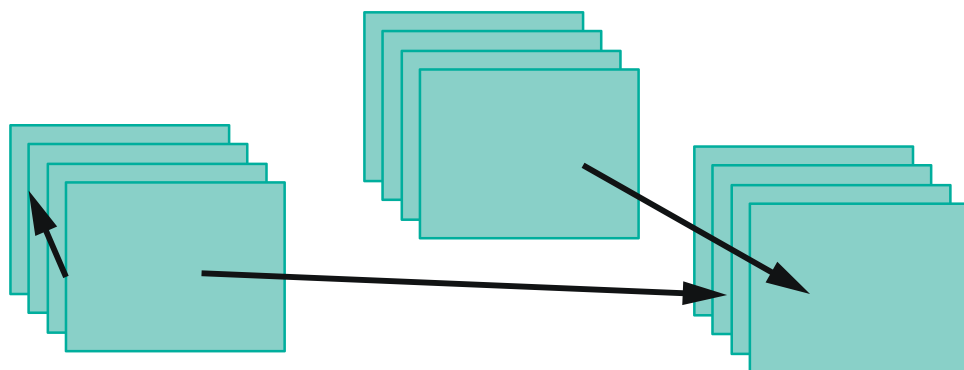


Fig. 2.15

El desarrollo de una aplicación puede concebir la navegación entre tarjetas de diferentes pilas.

¿Cómo se desarrolla una aplicación “LiveCode”?

Para la ejecución de esta aplicación se siguen los pasos siguientes:

1. Cuando se inicia la herramienta de desarrollo, ya sea a partir del icono de acceso directo o por su inicio en el menú de los programas, se tiene que crear un nuevo proyecto.

El nuevo proyecto se puede crear dando clic en la opción **New (Nuevo)** del **Start Center (Centro de Inicio)** (figura 2.16) o seleccionando en el menú **File (Archivo)** la opción **New Stack (Nueva pila)**, a continuación se opta por la variante **Default Size (Tamaño implícito)** y luego se adapta esta a nuestras necesidades (figura 2.17).

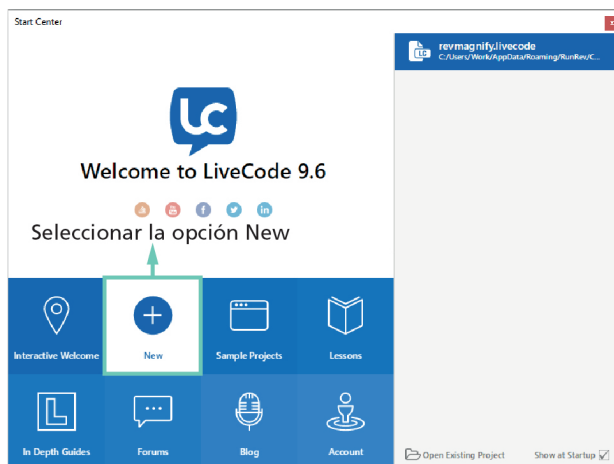


Fig. 2.16

Observación

Fíjate que **LiveCode** propone diferentes tamaños para la página o ventana principal, teniendo en cuenta los diferentes tipos de dispositivos móviles (teléfonos y tabletas).

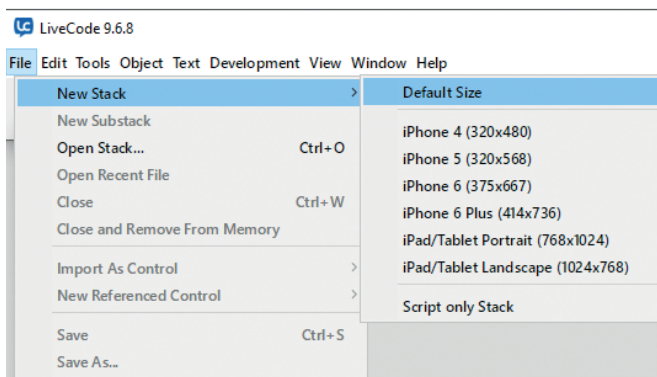


Fig. 2.17

El nuevo proyecto contendrá un “Main Stack” que será algo así como la **“página” principal**.

Automáticamente se crea una pila con una única tarjeta con el tamaño definido.

2. Se diseñan objetos como botones de diversos tipos, campos, figuras geométricas; se importan recursos multimedia como sonidos, video, en

diferentes tarjetas, con ayuda de la **Tools Palette** (paleta de herramientas) (figura 2.18).

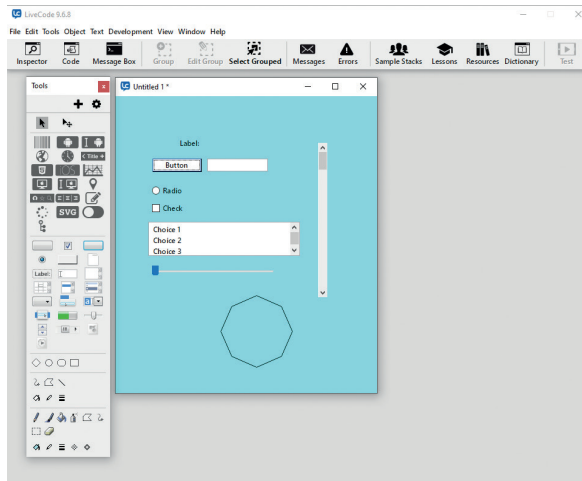


Fig. 2.18

3. Se programa la interactividad con los objetos necesarios.
4. Se define la navegación entre las tarjetas.
5. Se determina para que sistema operativo se trabaja.
6. Se compila y se obtiene el ejecutable de la aplicación.

Interfaz de LiveCode

Como ya se ha dicho, LiveCode es una herramienta de programación visual, eso quiere decir que la interfaz de desarrollo, también denominada IDE (**Interface Development Environment**) está conformada por una ventana principal y un **sistema de paletas** con botones reactivos y otros objetos que servirán para diseñar nuestra aplicación. Aunque el IDE o interfaz de desarrollo de LiveCode resulta relativamente sofisticado, en esta unidad nos referiremos a los aspectos mínimos que nos permitan implementar los algoritmos correspondientes a este nivel, reservando otros elementos para estudios futuros.

Tools Palette (Paleta de herramientas)

LiveCode posee una paleta de herramientas con la cual podrás diseñar una amplia variedad de objetos reactivos o interactivos. En este momento

se centrará especial atención en dos de estos objetos: el objeto **button** (botón) y el objeto **field** (campo) (figura 2.19).

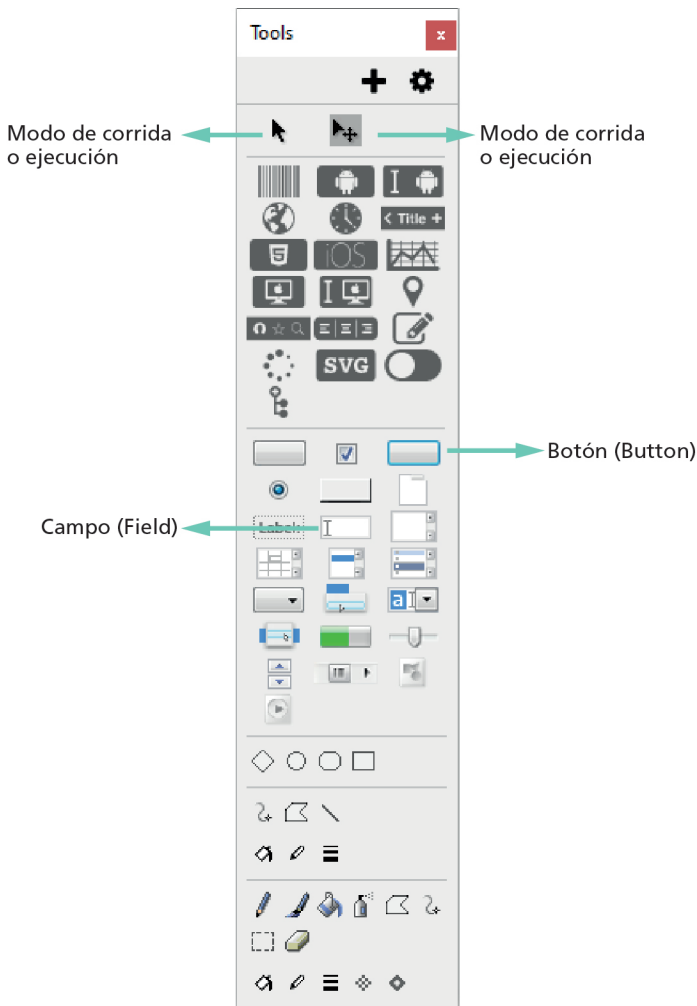


Fig. 2.19

En la figura 2.19 hay dos saetas en la paleta que permiten conmutar los modos de trabajo de LiveCode: modo de diseño o programación y el modo de corrida o ejecución.

Implícitamente LiveCode se presenta en modo de diseño o programación, lo cual le permite diseñar objetos y programarlos. Una vez realizado lo anterior se pasará a probar lo hecho para ver cómo funciona, entonces

se va a seleccionar la saeta de la izquierda (**Modo de corrida o ejecución**) y LiveCode pasará al modo usuario, lo cual permitirá comprobar lo hecho.

Esta concepción permite ir probando el desarrollo de la aplicación en cada momento que se estime conveniente.

2.3.2 Los eventos, los objetos y sus propiedades en LiveCode

Como se ha dicho con anterioridad, LiveCode es un lenguaje **basado en objetos** y **dirigido por eventos**, esto quiere decir que entre otras muchas cosas el flujo del programa va a estar determinado por un sistema de **acción y reacción**, o sea, que para que se ejecute un programa en LiveCode, tiene que ocurrir algo que desencadene las acciones del programa en calidad de respuesta. Eso que ocurre y desencadena acciones del programa se denomina **evento**.

Eventos típicos de los lenguajes visuales son, entre otros, los siguientes:

- Hacer clic izquierdo con el ratón sobre un botón u otro objeto (**MouseDown**).
- Oprimir teclas (**KeyDown**).
- Entrada del cursor del ratón sobre un objeto (**MouseEnter**).

Al producirse un evento por parte del usuario o por parte del propio sistema, se genera un mensaje que es interpretado por LiveCode y que en virtud del programa escrito, en el manipulador del evento se producirán las correspondientes acciones.

Es importante decir que los objetos son materializaciones o instancias de naturaleza abstracta llamadas **clases**, que no serán abordadas en este curso. Sin embargo, hay algo de los objetos que si se va a estudiar porque serán de gran utilidad, y son sus **propiedades**.

Las propiedades de los objetos

Las **propiedades de los objetos** son datos que conviven con los objetos y forman parte de estos. En la vida práctica esto también ocurre, por ejemplo, si tienes un objeto mesa, este va a tener un conjunto de propiedades inherentes a él, es decir, podrá ser cuadrada o redonda; podrá tener 4 o 3 patas, será azul o blanca; en fin habrá todo un sinnúmero de propiedades con sus respectivos valores que pueden caracterizar ese objeto. Por ello se podrá decir que la mesa es redonda, pero esa propiedad está caracterizada

por un valor numérico que puede ser un diámetro de 60 cm, o cuando se habla de sus patas también podríamos caracterizar la propiedad longitud de la pata de la mesa con un valor. En resumen, se podrá decir que las propiedades de los objetos determinan en ellos dos cosas:

- a. Su apariencia
- b. Su comportamiento

Por ejemplo, en la figura 2.20 se tiene un botón y claramente se lee en su leyenda o título la palabra **Solución**, por otro lado, están las etiquetas Punto 1 y Punto 2, dos objetos que muestran esos carteles, donde finalmente hay dos campos que poseen los valores 5,4 y 3,2.

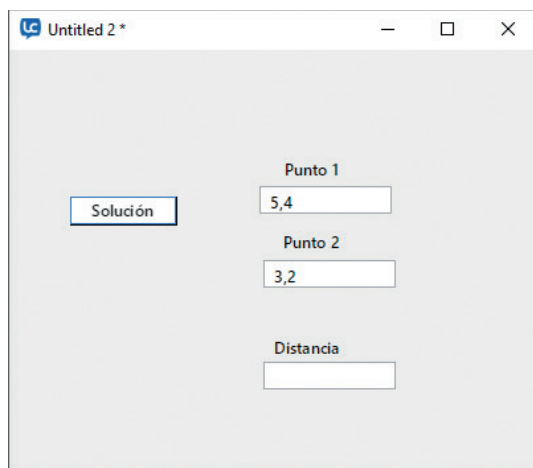


Fig. 2.20

Toda esta información se muestra en virtud de propiedades que poseen estos objetos.

¿Cómo acceder o modificar las propiedades de un objeto?

A las propiedades de un objeto se puede acceder de dos maneras:

- Mediante el inspector de propiedades.
- Mediante el lenguaje de programación.

El **inspector de propiedades** es un mecanismo del IDE o Interfaz de desarrollo que permite editar directamente las propiedades de un objeto. Para acceder basta con seleccionar el objeto, hacer clic derecho sobre él y

seleccionar la opción administrador de propiedades (*Property Inspector*), como se muestra en la figura 2.21.

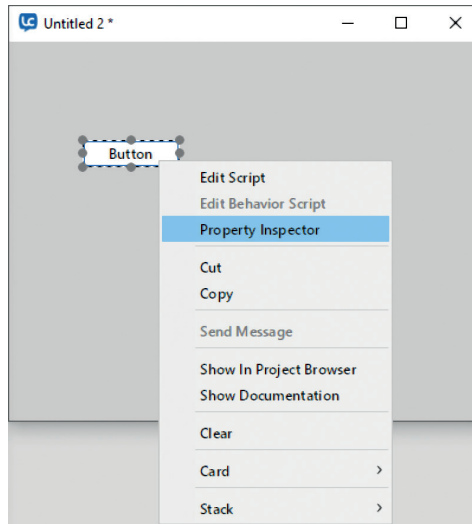


Fig. 2.21

Al acceder a la opción *Property Inspector*, aparecerá el siguiente cuadro de diálogo (figura 2.22).

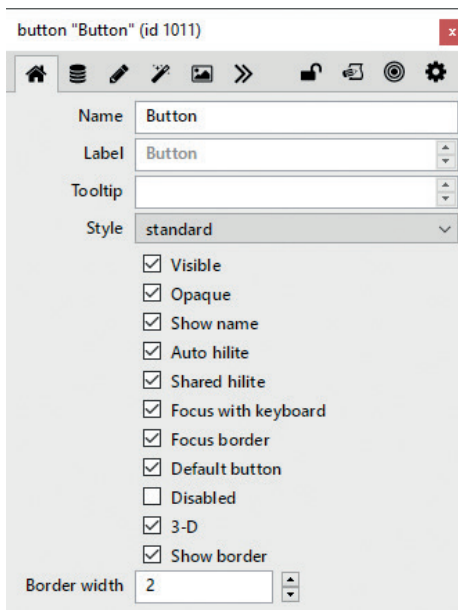


Fig. 2.22

Este cuadro de diálogo posee nueve secciones (figura 2.23) que invariablemente permiten acceder a diferentes propiedades de un objeto clasificadas por grupo, por ejemplo, el primer grupo son las *propiedades más generales* como el identificador del objeto, su etiqueta (texto que se muestra), visibilidad, opacidad, estilo de borde, etc.; el segundo grupo se refiere a lo que se denomina *Propiedades definidas por el usuario*, aspecto que será tratado más adelante; el tercer grupo está dedicado a los colores de texto, de fondo, texturas;

el cuarto trabaja los íconos que pueden acompañar a un botón; el quinto grupo se refiere al tamaño y la posición del objeto, el sexto al tipo de letra y tamaño, así como la alineación del texto. Los grupos restantes no serán estudiados en este grado.



Fig. 2.23

Es importante saber que para cambiar el contenido de las etiquetas con el rótulo Punto 1 y Punto 2, tienes que seleccionar un objeto de tipo **Label** (etiqueta) y cambiar la propiedad **Contents** (2do. grupo de propiedades), como se muestra en la figura 2.24.

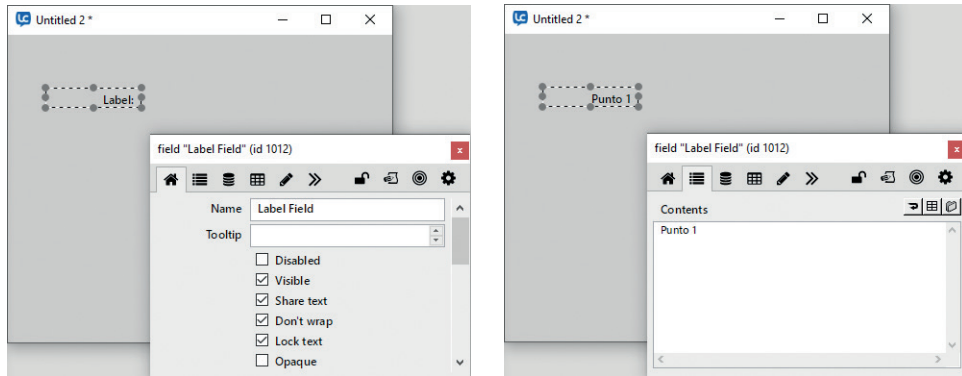


Fig. 2.24

La otra manera de modificar las propiedades de un objeto es mediante el uso del lenguaje de programación, para lo cual se usa la siguiente sintaxis:

Asignación de propiedades usando el lenguaje

Set the properties of Object to Valor
(Asigne a la propiedad del Objeto el valor)

Para referirnos a un objeto en LiveCode se usa la sintaxis: **Tipo + Nombre** o **Tipo + idNumber**.

Se puede suponer que hemos denominado **Solución** al botón al que nos referíamos en el ejemplo mostrado en la imagen 2.20 y quisiéramos

que este tuviera color azul, entonces a nivel del lenguaje de programación diríamos:

Set the background Color of button "Solución" to blue

2.3.3 Propiedades personalizadas o propiedades definidas por el usuario

El empleo de propiedades es una de las características de la programación orientada o basada en objetos, ya que una propiedad puede definir tanto la apariencia del objeto como su funcionalidad. LiveCode posee cientos de propiedades en sus objetos capaces de resolver disímiles problemas, pero ¿podemos nosotros definirle a los objetos de LiveCode nuevas propiedades? Satisfactoriamente la respuesta es positiva y el tema no resulta complejo.

Ya habíamos visto que la sintaxis para definir una propiedad es:

Set the properties of Object to Valor
(asignar la propiedad del objeto con el valor...)

Pues bien, LiveCode ha sido construido con la suficiente *inteligencia* para interpretar sentencias de este tipo: si en el lugar de la sintaxis, encuentra una propiedad que existe, le asigna el valor especificado; pero si no encuentra esta propiedad, "deduce" que la estamos creando y ¡automáticamente la crea y le asigna el valor especificado! Entonces se podría crear la propiedad "MiLargo" y asignarle el valor 255 de la siguiente manera:

Set MiLargo to field "Contenido" to 255

En tanto que la propiedad "Milargo" no existe y la sintaxis aplicada es correcta, LiveCode automáticamente crea la propiedad "MiLargo" y le asigna el valor 255. A partir de este momento el valor 255 forma parte del campo **Contenido** y este valor persistirá, o sea, perdurará mientras no sea eliminado, no es volátil como las variables, que una vez que apagues el sistema informático su valor desaparece.

Las *propiedades de usuario* son muy importantes porque:

- Permiten asociar datos a objetos con los que se relacionan (encapsulamiento).
- Brindan un nivel de acceso a los datos más rápido que con las variables.
- Protegen la información controlando la forma en que se lee o se escribe.

Para crear Propiedades de usuario desde la interfaz de LiveCode, puedes proceder de la siguiente forma (figura 2.25):

1. Seleccionar el objeto al cual se desea crearle la propiedad.
2. Dar clic derecho y acceder al Inspector de propiedades.
3. Seleccionar el panel **Custom Properties** (segundo ícono).
4. Entrar un nombre para la nueva propiedad de usuario (**Key**).
5. Entrar un valor (**Value**).

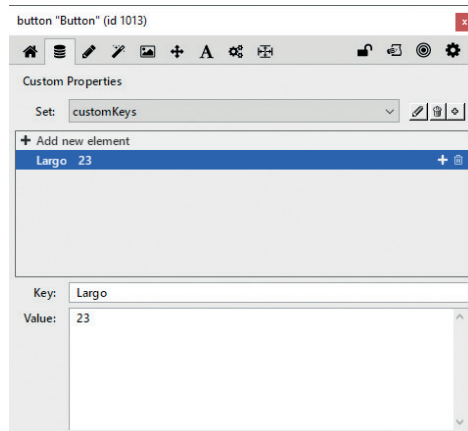


Fig. 2.25

Una vez asignada la propiedad de usuario a un objeto es posible recuperarla mediante la sentencia:

Put the Propiedad of Obj into Var

Si deseas saber qué propiedades de usuario posee un objeto, puedes seleccionar el objeto y usar la sentencia:

***get customkeys of the selectedobject
put IT***

2.3.4 Entrada y salida

A pesar de que LiveCode maneja más de 2 600 sentencias, bastarán dos de ellas (**Ask** y **Answer**), tres estructuras de control de flujo y los operadores estudiados, para implementar los algoritmos estudiados en este curso.

Instrucción Ask

Al emplear una **sentencia Ask** se mostrará una caja de texto (figura 2.26) que permitirá la **entrada** por escritura de cualquier tipo de dato (numérico o alfanumérico). Esta instrucción será nuestro equivalente a **Leer en pseudocódigo**.



Fig. 2.26

Instrucción Answer

Al emplear una **sentencia Answer** se mostrará una caja de texto que permitirá la **salida** de cualquier tipo de dato (numérico o alfanumérico). Esta instrucción será nuestro equivalente a **Escribir en pseudocódigo** (figura 2.27).

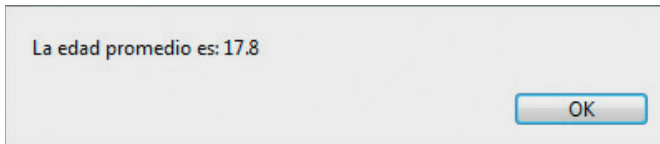


Fig. 2.27

Ejemplo 1

Describir un algoritmo que permita calcular el espacio recorrido por un cuerpo que se mueve con MRU (Movimiento Rectilíneo Uniforme) si se conocen la velocidad y el tiempo transcurrido en el desplazamiento. Al analizar este ejercicio te darás cuenta que el algoritmo de solución es de secuencia lineal.



Recuerda que...

En los **algoritmos de secuencia lineal**, o sea, en la ejecución de la secuencia, los pasos se aplican uno tras otro, en el orden en que aparecen en el algoritmo. Ese orden no se puede cambiar durante la ejecución, aun cuando sepamos que no se altera el resultado final al intercambiar dos de los pasos entre sí. En todo caso, lo que se sabrá es que puede haber otra variante para el mismo algoritmo donde aparezcan esos pasos intercambiados, sin que se altere el resultado.

Aquí correspondería escribir el pseudocódigo, que en este caso sería:

Algoritmo: calcular distancia

Inicio

Leer V

Leer t

*S := V * t*

Escribir S

Fin

Ahora para poder implementar este algoritmo se seguirán los pasos siguientes:

Solución

1. Se abre LiveCode y desde la Paleta de herramientas (**Tools Palette**), se pone el **Modo de edición**.
2. Se diseña un botón (para crear el botón debes seleccionarlo en la Paleta de herramientas y arrastrarlo hasta el lugar donde lo deseas ubicar).
3. Se hace clic derecho sobre el botón para pasar a programarlo y a continuación se selecciona la opción **Edit Script** (figura 2.28).

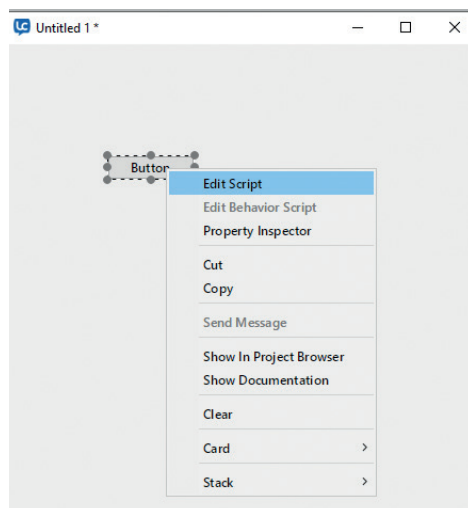


Fig. 2.28

4. Aparece el Editor de código con el **Manipulador de eventos (Handler list)**, el Área para escribir el código y la descripción de errores, entre otras opciones, como se muestra en la figura 2.29. Aquí seleccionarás el evento **MouseUp** (hacer clic con el botón izquierdo del ratón) y aparecerá en el Área de código el evento, y el cursor estará entre el inicio y el fin del evento para que puedas escribir el programa, como se muestra en la figura 2.30.

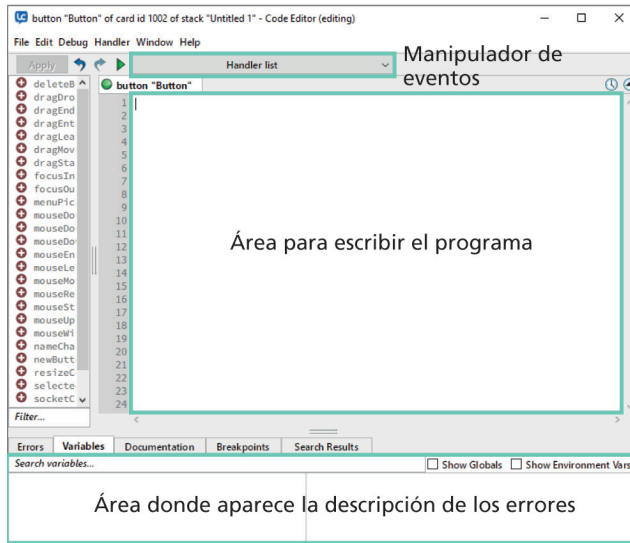


Fig. 2.29

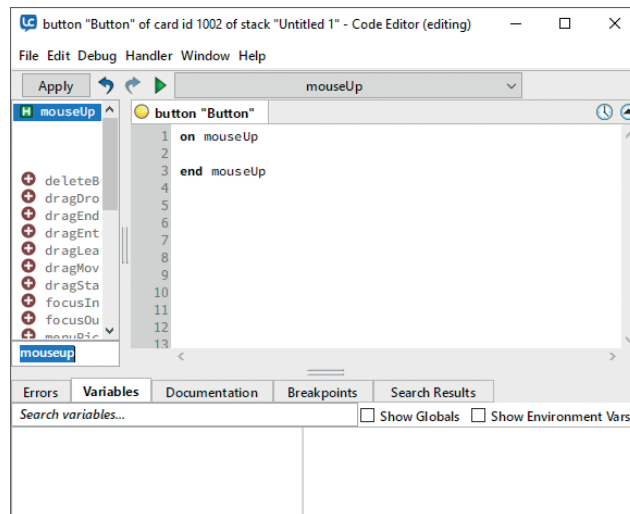


Fig. 2.30

5. Se escribe el programa:

Se recordará primero el *pseudocódigo*:

Inicio

Leer V

Leer t

*S:= V*t*

Escribir S

Fin

Este sería el programa en LiveCode (figura 2.31).

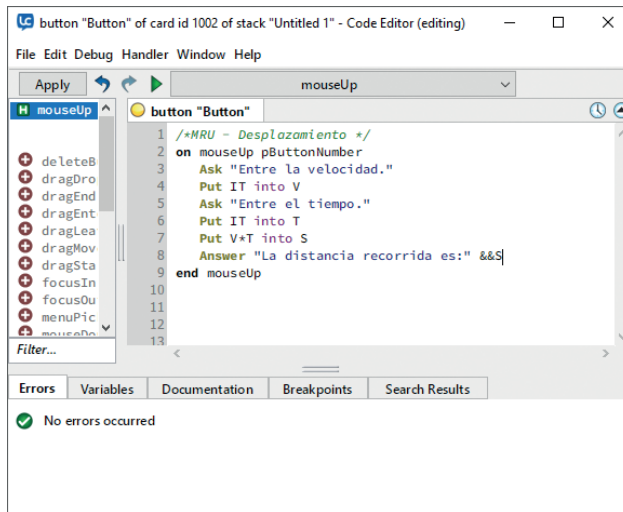


Fig. 2.31

Analicemos ahora el código:

1. Doble guion (--) significa el comentario para poner el título del programa o comentar lo que hacen las instrucciones del código. También se puede escribir /* y cerrar con */ para comentarios de varias líneas:
- Ask: sentencia para la lectura de datos. La sentencia **Ask** deposita el dato automáticamente en una variable del sistema denominada **IT**. Como el próximo **Ask (del tiempo)**, sobrescribirá el dato de **la Velocidad**, se hace necesario asignar el valor de **IT** a una variable independiente **V**. **Put IT into V**. De igual manera se procederá con el tiempo **PUT IT into T**.

- Luego se asigna el producto de $V \cdot T$ a la variable S y finalmente se escribe el resultado con ayuda de la sentencia **Answer**:
Answer "La distancia recorrida es:" && S

El **&** es el símbolo de concatenación que se expresa en **pseudocódigo** como **UnE**, cuya función es unir o concatenar el mensaje "La distancia..." con el valor S .

Ejemplo 2

Al interactuar con el interruptor de la luz del comedor, se constata que la luz no enciende. Implemente un programa que simule esta situación. Al analizar este ejemplo te darás cuenta que en la solución emplearás un algoritmo con alternativa simple y una estructura de control.



Recuerda que...

Existen dos tipos de alternativas: las simples y las múltiples. Las **alternativas simples** comprueban el valor veritativo de solo una condición. Las **múltiples**, por su parte, verifican el valor veritativo de varias condiciones.

Solución (figura 2.32):

```
/* Bombillo */
On MouseUp
  answer "Se busca una escalera"
  answer "Se sube hasta la altura del bombillo"
  answer "¿Bombillo flojo?" with Sí or No
  if IT is Sí then
    answer "Apretar bombillo"
  else
    answer "Cambiar bombillo"
  end if
end MouseUp
```

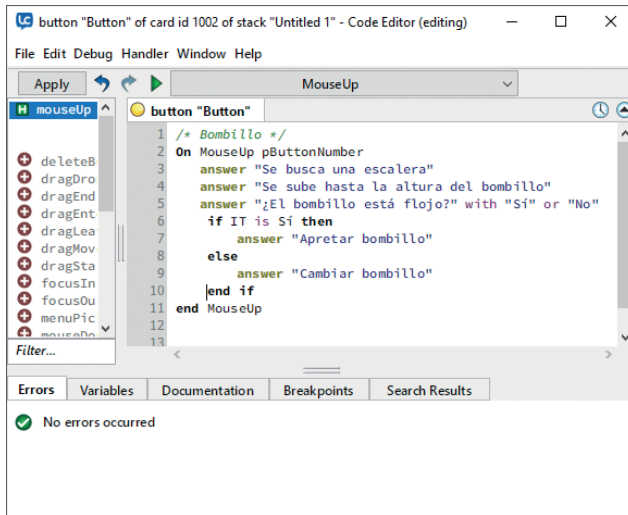


Fig. 2.32

Ahora analicemos el código:

1. La situación planteada en el ejemplo 2 no posee un modelo matemático explícito (no hay fórmulas ni ecuaciones).
2. Como te habrás fijado, la primera parte del código es igual al ejemplo anterior, ya que se utiliza el mismo evento (*mouseUp*).
3. La tercera línea de código emplea una variante especial de la sentencia **Answer** mediante la palabra reservada **with**. Esto significa que al mostrar el mensaje se propondrá una interrogante en forma de **Sí** o **No**, lo cual bifurcará el flujo del programa en dos direcciones, una vez para **Sí** y otra vez para **No**.
4. Siendo estrictos, las cadenas **Sí** y **No** deberían escribirse entre comillas "Sí" y "No", pero el carácter débilmente tipado del lenguaje tolera la no contemplación estricta de esta regla en tanto estas expresiones no coincidan con variables cuyas denominaciones sean por coincidencia Sí y No. **No obstante una buena práctica es poner las cadenas entre comillas.**
5. Como ya se explicó tanto **Ask** como **Answer** asignan el dato introducido en la variable del sistema denominada **IT**.
6. La sentencia **if – then – else – end if** permite implementar los algoritmos de tipo **alternativos**.



Recuerda que...

La sintaxis de la sentencia **if – then – else – end if** es:

If condición **then**

Acciones

else

Acciones

End if

Ejemplo 3

Dado un conjunto de 10 números enteros, determinar cuántos números son pares. Al analizar este ejemplo te darás cuenta que en la solución emplearás un algoritmo con un ciclo, por lo que será necesario utilizar una estructura de control.



Recuerda que...

Los **algoritmos con ciclos** son aquellos donde aparecen con frecuencia situaciones en las que se necesita que un conjunto de instrucciones se ejecute repetidamente. En este caso, se debe contar con estructuras de control que permitan la iteración o repetición y la verificación de condiciones que determinan si el conjunto de instrucciones debe seguir siendo repetido o si debe detenerse.

Solución (figura 2.33):

```

/* Cantidad de enteros */
On MouseUp
put 0 into C ---- Inicialización del contador
Repeat 10 times -- Estructura cíclica
ask "Entre un número:" --- Entrada del dato
if IT mod 2 = 0 then --- Alternative de número par
add 1 to C -- Increment de contador
end if
end repeat --- Fin de la estructura cíclica
Answer "La cantidad de números enteros es " & C -- Respuesta
end MouseUp

```

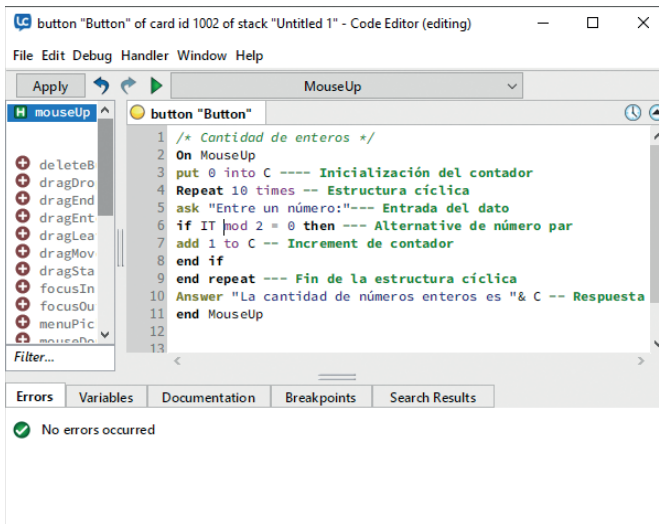


Fig. 2.33

Ahora analicemos el código:

1. El inicio del código sigue siendo el mismo pues estamos utilizando el mismo evento (*MouseUp*).
2. **put 0 into C**, esta sentencia inicializa el contador C que se empleará para contar los números que sean pares. Recuerda que tanto los contadores como los acumuladores deberán ser inicializados antes de ser usados.
3. **Repeat 10 times / end repeat** (esta sentencia permite realizar un ciclo, o sea, una iteración o repetición de acciones, tantas veces como se especifique, en este caso 10 veces.
Nota: **Repeat – End repeat**- exige que se conozca a priori la cantidad de veces que se producirá el ciclo de acciones.
4. **IT mod 2 = 0** (IT es la variable del sistema en que la sentencia Ask depositó el dato entrado, $IT \text{ mod } 2 = 0$ comprueba si la división del número que se encuentra en la variable IT entre 2 da como residuo 0, lo cual significa que el número que se encuentra en la variable IT es par.
5. **add 1 to C**, es una manera abreviada de implementar un contador, un sinónimo de esta expresión contadora podría ser **Put C + 1 into C**.
6. El operador **&** como ya se ha visto antes, es el operador de concatenación de cadena de caracteres para producir un mensaje combinando texto y resultado.

Comprueba lo aprendido

1. ¿En qué elementos radica la dificultad de resolver problemas?
2. ¿Qué tres momentos son invariantes en la solución de problemas?
3. ¿Qué plantea el teorema fundamental de la programación estructurada?
4. ¿A qué se llama propiedades de objetos?
5. ¿De cuántas maneras se pueden modificar las propiedades de un objeto usando LiveCode?
6. Menciona algunas secciones del cuadro de diálogo de propiedades de objetos en LiveCode.
7. ¿Cuál es la sintaxis general para asignar un valor a una propiedad?
8. ¿Qué es una propiedad de usuario y cómo se define?
9. ¿Cuál es la importancia de las propiedades de usuario?
10. ¿Qué es una lista en LiveCode?
11. ¿Cómo se logra obtener un elemento de una lista en LiveCode?
12. ¿A qué se llama **evento** y qué diferencia o similitud tiene este concepto con el de **mensaje**?
13. Menciona cinco eventos y sus respectivos mensajes manejables por LiveCode.
14. Explica con tus palabras la jerarquía de mensajes de LiveCode.
15. ¿En qué nivel de jerarquía se debe colocar un manipulador de mensajes?

16. ¿Cuál es la sintaxis para definir un mensaje personalizado o definido por el programador?
17. Qué relación guardan los conceptos de función en Matemática y en Informática.
18. Menciona tres funciones **build-in** de LiveCode.
19. ¿Cuál es la sintaxis para definir una función creada por el programador?
20. A qué se llama variable global y cómo se define.
21. ¿Cómo se pueden almacenar varios datos en una misma variable?
22. ¿Por qué se dice que LiveCode es un Lenguaje dirigido por eventos?
23. Estudia los diferentes eventos del mouse, manejables por LiveCode, y caracteriza a cuatro de ellos.
24. Elabora un cuadro resumen sobre los diferentes controles (objetos de control) que se pueden utilizar en LiveCode que tenga la estructura que se muestra en el ejemplo siguiente:

Control	Función
Botón	Realiza una acción o serie de acciones específicas.

25. Analiza los siguientes algoritmos y ponle un nombre adecuado:

a) /**/

On MouseUp

answer "Me aproximo al borde de la acera."

answer "Miro a la izquierda."

answer "Miro a la derecha."

answer "¿Viene carro?" with Sí or No

```

if IT is Sí then
  answer "Me detengo."
else
  answer "Cruzo."
end if
end MouseUp

```

```

b) /**/
On MouseUp
Ask "Entre el valor de la velocidad inicial."
Put IT into VInic
Ask "Entre el valor del tiempo transcurrido."
Put IT into VTiempo
Ask "Entre el valor de la aceleración."
Put IT into VAcel
put  $VInic * VTiempo + (VAcel * VTiempo^2) / 2$  into EspacioRecorrido
Answer EspacioRecorrido
end MouseUp

```

```

c) /**/
On MouseUp
Ask "Entre un número"
Put IT into N
if N > 0 then
  answer N
else
  answer -1 * N
end if
end MouseUp

```

26. En el siguiente programa existen cinco errores. Encuéntralos.

```

On MouseUp
Put field Coeficientes into Lista
Put item 1 of Lista into a
Put item 2 of Lista into b

```

```

Put item 3 of Lista into c
Put  $b^2 - 4*a + c$  into Delta
Switch
Case Delta > 0
Answer "Dos raíces reales"
Put  $-b + \sqrt{\text{Delta}} / 2*a$  into X1
Put  $b + \sqrt{\text{Delta}} / 2*a$  into X2
Answer "Raíz = " & X1 & "Raíz = " & X2
break
Caso Delta = 0
Answer "Una raíz doble"
put  $-b / 2*a$  into X1
Answer "Raíz =" & X1

default
Answer "No hay raíces reales"
End switch
End

```

- 27.** Resuelva los siguientes problemas codificándolos en LiveCode:
- Escriba un programa que convierta de centímetros a pulgadas. Una pulgada es igual a 2,54 cm.
 - Escriba un programa que determine si el número entero ingresado por el usuario es par o no.
 - Escriba un programa que pida dos números enteros y que calcule la división, indicando si la división es exacta o no.
 - Escriba un programa que pida al usuario dos palabras, y que indique cuál de ellas es la más larga y por cuántas letras lo es.
 - Escriba un programa que reciba como entrada dos números, y los muestre ordenados de menor a mayor.
 - Escriba un programa que simule una calculadora básica, este puede realizar operación de suma, resta, multiplicación y división. Debe recibir como entrada dos números reales y un operador, que puede ser +, -, * o /. La salida del programa debe ser el resultado de la operación.

BIBLIOGRAFÍA

- AMENGUAL ARGUDO, J. C., T. LÓPEZ y VICENTE RAMÓN: "Informática Básica. Conceptos básicos de informática", Publicacions de la Universitat Jaume, recuperado de www.sapientia.uji.es
- APARICI, R. y A. GARCÍS MANTILLA: *Lectura de imágenes en la era digital*, Ediciones de La Tore, Madrid, 2008.
- BASOGAIN OLABE, XABIER: "Pensamiento computacional a través de la programación: paradigma de Aprendizaje", www.um.es/ead/red/46/Basogain.pdf, 2015
- BUDD, T.: *Introducción a la programación orientada a objetos*, Ed. Addison-Wesley Iberoamericana, Madrid, 1994.
- CANGALAYA SEVILLANO, J. A.: *Estrategias de aprendizaje de la metodología activa*, Grupo de Capacitación Pedagógica, Lima, 2010.
- CHUN, B. y T. PIOTROWSKI: "Pensamiento computacional ilustrado", www.ctil-lustrated.com/
- Colectivo de autores: "Pensamiento computacional. Un aporte para la educación de hoy", Montevideo, Uruguay, www.gurisesunidos.org.uy
- Fundación misión Sucre: *Ejemplos y ejercicios. Algoritmia*, Ministerio de Educación Superior de la República Bolivariana de Venezuela, Venezuela, [s. f.].
- GARCÍA M. E.: "Introducción a la Informática. Ejercicios resueltos de algoritmos", www.proformatiasgarcia.com.ar
- GARRIGA GARZÓN, F.: *Problemas resueltos de programación lineal*, Ed. Omnia Publisher, versión digital.
- GOLDBERG, S.: *LiveCode Lite: Computer Programming Made Ridiculously Simple*, Ed. MedMaster Inc, Miami, FL., [s. a.].
- HOLGATE, C. y J. GERDEEN: "LiveCode Mobile Development Beginner's Guide", (segunda), Packt Publishing Ltd, www.packtpub.com

- JOYANES AGUILAR, L.: *Fundamentos de programación. Algoritmos y Estructuras de datos y objetos*, Ed. McGraw-Hill, Madrid, 2003.
- LAVIERI E.: "LiveCode Mobile Development Cookbook", primera parte, Ed. Packt Publishing Ltd, Birmingham–Mumbai, www.packtpub.com, 2014.
- LÓPEZ CRUZ, P. A.: "Manual fundamental. *Hardware* y componentes", Ed. Anaya, [s. e.], [s. a.].
- LÓPEZ GARCÍA, J. C.: "Algoritmos y programación. Guía para docentes", www.eduteka.org, 2009.
- MUÑOZ, A.: "La imagen fotográfica", Ministerio de Educación, Política Social y Deporte, Centro Nacional de Información y Comunicación Educativa, Madrid, en <http://w3.cnice.mec.es/eos/MaterialesEducativos/mem2006/fotografia/index.html#>
- NOGUEZ MONROY, J.: *Informática*, segunda parte, Ed. Santillana, México, 2006.
- OSUNA GUERRERO, R. y E. GARCÍA CERRO: "Fundamentos de fotografía digital", UNED, Madrid, en <http://ocw.innova.uned.es/ocwuniversia/tecnologia-electronica/fundamentos-de-fotografia-digital>.
- PÉREZ NARVÁEZ, HAMILTON OMAR: "Herramientas informáticas para el desarrollo del Pensamiento computacional", versión digital.
- QUEREDA GÓMEZ, E. D., y L. BAÑÓN BLÁZQUEZ: "Colección de exámenes de la asignatura Aplicación de Ordenadores 2005–2011. Fundamentos de programación y algoritmia", versión digital.
- RODRÍGUEZ RODRÍGUEZ, L. A.: *Revolución multimedia*, ISP Felix Varela: Centro de estudios Visofted, [s. a.].
- STALLINGS, W.: *Organización y arquitectura de computadores*, séptima edición, Ed. Pearson Educación, S. A., Madrid, 2005.
- TAMAYO SILVA, F.: *Lógica de Programación y Lenguajes*, Instituto Superior Pedagógico Pepito Tey, Las Tunas, 2002.
- ZAPATA OSPINA, C. A.: *Fundamentos de programación. Guía de autoenseñanza*, (primera), RA-MA editorial, Colombia, 2006.
- ZAVALETA LUNA, D. M. y J. J. FRANCISCO LEÓN: *LiveCode para Android a través de ejemplos: nivel básico*, (primera), Universidad Juárez Autónoma de Tabasco, México, www.pensamientocomputacional.org

