

UNIVERSIDAD NACIONAL DE LOMAS DE ZAMORA



FACULTAD DE INGENIERÍA INGENIERÍA MECATRÓNICA



FUNDAMENTOS DE LOS COMPUTADORES DIGITALES MÓDULO 1 - CURSO 2024

AUTORES

Diego Servetto
Ezequiel Blanca
Néstor Maceda

- Lic. en Informática - Esp. en Gestión Tecnológica (UNLZ)
- Ingeniero en Electrónica (UTN)
- Licenciado en Informática Educativa (UNLa)



License Creative Commons

ÍNDICE

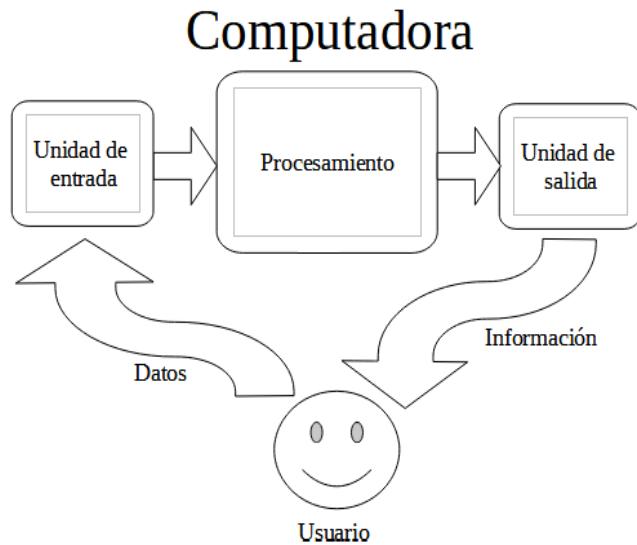
INTRODUCCIÓN A LOS COMPUTADORES	5
Definición de computadora	5
Diferencia entre calculadora y computadora	5
Tipos de computadores	6
Computador analógico	6
Computador digital	6
ENIAC (Electronic Numerical Integrator and Computer)	6
EDVAC (Electronic Discrete Variable Automatic Calculator)	7
ARQUITECTURA DE VON NEUMANN	7
Elementos principales	7
Funcionamiento básico	7
Característica fundamental	7
COMPUTADORAS DIGITALES MODERNAS	8
Hardware	8
Software	8
INTRODUCCIÓN AL SOFTWARE	9
Tipos de software	9
Software de Sistema	9
Software de Aplicaciones	9
Software de Seguridad	9
Software de desarrollo	9
Software embebido	10
ASPECTOS DEL DESARROLLO DE SOFTWARE	10
ALGORITMO	11
Definición de algoritmo	11
Ejemplos de algoritmos	11
Algoritmo Culinario	11
Algoritmo Mecánico	12
Propiedades de un algoritmo	12

Propiedades principales	12
Otras propiedades	13
Tipos de algoritmos	13
Según su diseño	13
Según su complejidad	13
Según su función	13
Expresión de un algoritmo	14
Lenguajes informales	14
Lenguajes formales	14
Implementación de un algoritmo	17
Desarrollador de algoritmos	18
 PSEUDOCÓDIGO	 19
Definición	19
Sintaxis	19
Parte formal	19
Parte informal	20
Ejemplo	21
 DIAGRAMA DE FLUJO	 22
Definición de diagrama de flujo	22
El diagrama para procesamiento electrónico	22
Estandarización ANSI	22
Simbología para la diagramación	22
Símbolos obsoletos	23
Símbolos fundamentales	24
Símbolos especializados	25
Ejemplos de diagramas de flujo	26
Ejemplo 1: Mi primer diagrama	26
Ejemplo 2: Entrada y salida	26
Ejemplo 3: Sumar dos números	27
Ejemplo 4: Verdadero ó Falso	27

INTRODUCCIÓN A LOS COMPUTADORES

Definición de computadora

La **computadora**¹ (del inglés: *computer* y este del latín: *computare*, ‘calcular’), también denominada **computador** u **ordenador** (del francés: *ordinateur*), es una **máquina** que ejecuta una serie de **operaciones complejas** para procesar **datos** de unidades de entrada, obteniendo convenientemente **información** que posteriormente se envía a las unidades de salida.



Diferencia entre calculadora y computadora

Si bien una computadora tiene su origen en el término de calculadora, se diferencia de este último por las siguientes características:

- La calculadora es de **propósito específico** y la computadora es de **propósito general**.
- La calculadora procesa **pocos datos** y una computadora procesa **grandes cantidades**.
- La computadora es una máquina de **mayor dimensión**, recursos y consumo que una calculadora.
- La computadora es **programable**, y la calculadora no, o bien es programable dentro del mismo propósito específico por el cual fue construida.

¹ **Definición:** debido a la evolución de la tecnología el concepto de computadora fue cambiando a lo largo del tiempo. Por ello, se buscó la definición lo más general posible que abarque la mayor cantidad de aspectos históricos, actuales y futuros.

Tipos de computadores

Computador analógico

Un computador analógico o computador real, es una **máquina**², en la cual los datos de entrada son valores físicos del contexto, que son percibidos en forma directa. Realiza operaciones sobre **valores reales** y diferenciales. La información se brinda sobre instrumentos de osciloscopios, amperímetros, etc, con valores reales aproximados. La programación se realiza a través de un cableado sobre un tablero de conectores. Por ello, al cambiar el problema a resolver, cambiaba el circuito de programación.



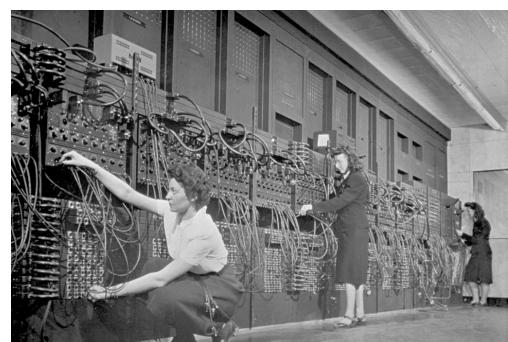
Los computadores analógicos se utilizaban para supervisar las condiciones del mundo real, tales como Viento, Temperatura, Sonido, Movimiento, etc y se caracterizaban por ser mucho más rápidos que las computadoras digitales para tal fin.

Computador digital

Un computador digital es una máquina³ (conjunto de aparatos⁴ combinados) que tiene por objeto el procesamiento de datos⁵ a través de mecanismos basados en operaciones de **valores discretos** o de estados finitos, en el campo de los **números naturales**. Ejemplos:

ENIAC (Electronic Numerical Integrator and Computer)

Una de las primeras **computadoras digitales** con tecnología electrónica de válvulas y de propósito general fue creada por la Universidad de Pensilvania y puesta en funcionamiento en 1946. Operaba con **dígitos decimales (0 al 9)** y realizaba cálculos de hasta 20 cifras.



² **Máquina:** Conjunto de aparatos combinados para recibir cierta forma de energía y transformarla en otra más adecuada, o para producir un efecto determinado.

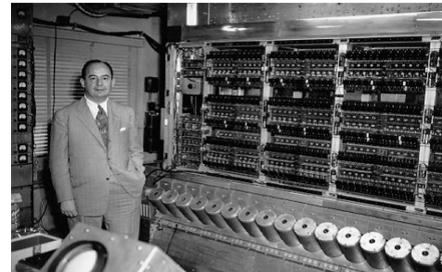
³ **Máquina:** Conjunto de aparatos combinados para recibir cierta forma de energía y transformarla en otra más adecuada, o para producir un efecto determinado.

⁴ **Aparato:** Conjunto de piezas que cumplen una función determinada.

⁵ **Procesamiento de datos:** Aplicación sistemática de una serie de operaciones sobre un conjunto de datos, generalmente por medio de máquinas, para explotar la información que estos datos representan.

EDVAC (Electronic Discrete Variable Automatic Calculator)

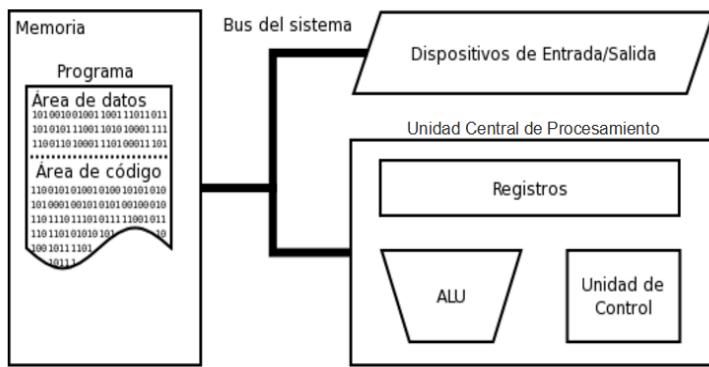
Comienza a operar en 1951, concebida por John Von Neumann, que a diferencia de la ENIAC las operaciones la realiza en el sistema **binario (0 y 1)**, y por primera vez el programa (conjunto de instrucciones) fue diseñado para ser almacenado en la misma computadora.



ARQUITECTURA DE VON NEUMANN

Elementos principales

La arquitectura o modelo del Ingeniero y matemático Von Neumann, es el modelo⁶ en el cual se basan las **computadoras digitales en la actualidad**. Internamente están compuestas por tres elementos principales como se observa en la figura. Un bloque que indica los dispositivos de entrada y salida: el cual son en realidad dos unidades, **UE** (unidad de entrada) y **US** (unidad de salida), luego un bloque de memoria o **UM** (unidad de memoria) donde se guarda el programa y los datos y finalmente un bloque **UCP** (unidad central de procesamiento) compuesta por una **UAL** (unidad aritmética lógica) o ALU (en inglés) un bloque de **registros** y una **UC** (unidad de control).



Funcionamiento básico

Los datos e instrucciones que son ingresados por la **UE**, son almacenados en la **UM** para ser procesados por la **UAL** en colaboración con los **Registros** y supervisión de la **UC**. La información convenientemente obtenida se le da salida por la **US**.

Característica fundamental

La principal característica de este modelo es que los **datos y las instrucciones se almacenan en una memoria interna**. Antes no se almacenaban los datos ni las instrucciones. Los datos e instrucciones eran externos al sistema y se introducían por medio de cableado (parte del circuito), tarjetas o cintas perforadas en el momento de ser utilizados.

⁶ http://academicos.acz.uam.mx/oan/lac/Murdoch_es.pdf

COMPUTADORAS DIGITALES MODERNAS

Hardware

Es la parte física o **tangible** de la computadora digital. Y está compuesta principalmente por: los **periféricos de entrada** (el teclado, el mouse, el micrófono, el escáner), los **periféricos de salida** (el monitor, los altavoces, la impresora), los **periféricos de comunicaciones** (modem, router) y el **gabinete principal** que incluye la Unidad central de procesamiento, la placa mother, las **unidades de almacenamiento**: memoria RAM (memoria volátil de acceso aleatorio), la memoria ROM (memoria de solo lectura), el discos rígido, etc. La **capacidad de almacenamiento del hardware** se puede medir mediante la unidad de información de las computadoras digitales, el bit.

Bit⁷, es el acrónimo de Binary Digit (Dígito binario) y es la unidad elemental de información. . Como una computadora puede almacenar y transmitir grandes cantidades de bit no se emplea la unidad básica sino múltiplos del mismo: 1 Byte = 8 bits.

HARDWARE



Unidades de Medidas de Almacenamiento

Medida	Simbología	Equivalencia	Equivalente en Bytes
byte	b	8 bits	1 byte
kilobyte	Kb	1024 bytes	1 024 bytes
megabyte	MB	1024 KB	1 048 576 bytes
gigabyte	GB	1024 MB	1 073 741 824 bytes
terabyte	TB	1024 GB	1 099 511 627 776 bytes
Petabyte	PB	1024 TB	1 125 899 906 842 624 bytes
Exabyte	EB	1024 PB	1 152 921 504 606 846 976 bytes
Zetabyte	ZB	1024 EB	1 180 591 620 717 411 303 424 bytes
Yottabyte	YB	1024 ZB	1 208 925 819 614 629 174 706 176 bytes
Brontobyte	BB	1024 YB	1 237 940 039 285 380 274 899 124 224 bytes
Geopbyte	GB	1024 BB	1 267 650 600 228 229 401 496 703 205 376 bytes

Software

Es un conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas. Es la parte **intangible** (la que no se puede tocar) son los programas y los datos que se almacenan en la memoria o disco de la computadora. Estos programas o software se almacena en forma binaria. Es decir en forma de ceros y unos. (por ejemplo 1101 0101).

SOFTWARE



⁷ En 1948, Bell Laboratories anunció la creación del **bit**, una unidad para medir la información. Su inventor fue el **Ingeniero Claude Shannon** de 32 años.

INTRODUCCIÓN AL SOFTWARE

En un principio, el estudio y el desarrollo de las computadoras estaba centrado en el hardware, por ello, era el campo de la ingeniería mecánica eléctrica y posteriormente electrónica. Como las primeras computadoras se utilizaban para hacer cálculos, los primeros programas eran desarrollados por matemáticos que se especializaban en este campo. Las computadoras eran una tecnología dura.

Posteriormente, con la aparición de las computadoras digitales, el programa se independizó del hardware y tomó importancia y gran envergadura, adquiriendo la propiedad de software (grandes paquetes de programas integrados y complejos) y su desarrollo requirió especialistas en esta disciplina, implementando las conocidas carreras de ingeniería de software, informática o sistemas.

Tipos de software

Software de Sistema

Es el software de base, o software operativo, es el que permite operar la computadora, es el que permite el control y funcionamiento del hardware, incluyendo todos sus periféricos. Sin este software la computadora digital no funciona.

Software de Aplicaciones

Son los software o programas que permiten darle una aplicación informática a las distintas disciplinas profesionales. Ejemplo de software de aplicación son:

- **Autodesk AutoCAD**, empleado por los arquitectos para hacer planos de edificios
- **Autodesk Inventor**, empleado por los ingenieros en diseños industriales.
- **Corel Draw**, empleado por los diseñadores gráficos para trabajar en publicidad
- **Proteus**, empleado por los ingenieros electrónicos para hacer circuitos.
- **Paquete office**, empleado por las pymes en oficinas para administrar y gestionar
- Etc..

Software de Seguridad

Es uno de los tipos de software más modernos que debieron implementarse a medida que las computadoras ganaron terreno en todas las operaciones sociales. Los malvivientes aprovecharon las computadoras digitales como herramienta para cometer delitos junto al uso de internet. Por lo que se debió resguardar la privacidad de las personas, cuidando sus datos sensibles como fotos y videos personales, claves de acceso, etc. Este software se conoce como sistemas antivirus.

Software de desarrollo

Es un software que se utiliza para desarrollar aplicaciones. Es un programa que hace programas.

Estos software se conocen como IDE (Integrated Development Environment), Entornos de desarrollo de software. Ejemplo de estos son:

- Microsoft Visual Studio Code
- Eclipse
- NetBeans
- Codeblock
- etc.

Software embebido

Son programas con aplicaciones específicas, generalmente del tipo industrial diseñado para realizar una o algunas pocas funciones dedicadas. Actualmente, todos los equipos tanto industriales (tornos fresadoras, robótica, control de procesos de producción, etc.) como domésticos (lavarrropas automáticos, microondas, heladeras, etc) poseen sistemas electrónicos digitales programados. Es decir, tienen un programa embebido o incrustado para realizar funciones especializadas. Este tipo de programas requiere un conocimiento acabado del hardware y es desarrollado principalmente por **ingenieros electrónicos o mecatrónicos** ya que trabajan con lenguajes de bajo nivel o de código máquina. Son sistemas operativos de “pequeñas computadoras” (microcontroladores) para máquinas especializadas.

ASPECTOS DEL DESARROLLO DE SOFTWARE

Todo desarrollo de software comienza con un **algoritmo** (un concepto abstracto) que luego es implementado en computadoras digitales a través de **programas** en un determinado **lenguaje de programación**. Hay muchos lenguajes de programación C, C++, C#, JAVA, PYTHON, ETC.. La conveniencia de uno u otro lenguaje de programación a utilizar dependerá del tipo de algoritmo.(la idea).



Podríamos decir haciendo analogías, que el desarrollo de un software se inicia con una idea, un concepto (**algoritmo**) y a través de un proceso, se implementa (**programa**) la misma en alguna tecnología (**lenguaje de programación seleccionado**) dando como resultado una aplicación (**software**) concreta en el mundo tangible.

ALGORITMO

Definición de algoritmo

Un algoritmo es un **conjunto ordenado y finito de operaciones** que permite hallar la solución de un problema determinado. El término algoritmo proviene del latín,dixit algorithmus y este del griego arithmos, que significa **número**, quizá también con influencia del nombre del matemático persa Al-Juarismi. Podría decirse que los algoritmos tuvieron su origen en la **matemática**, ejemplo famosos de ello son:

- Algoritmo de Euclides (300 AC): calcula el máximo común divisor
- Algoritmo de Dijkstra: para determinar el camino más corto en un grafo.
- Algoritmo de la Criba de Eratóstenes: obtener los números primos menores a un x número,

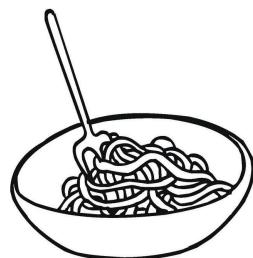
Sin embargo, la presencia de algoritmos no solo se limita a la matemática. Su uso se observa en todas las disciplinas e incluso en la vida cotidiana.

Ejemplos de algoritmos

Algoritmo Culinario

Un ejemplo interesante, es una receta de cocina con sus ingredientes y pasos para preparar spaghetti. Como dichos pasos, son efectivamente un conjunto ordenado de operaciones para alcanzar un resultado, estaríamos en presencia de un algoritmo aplicado al arte culinario.

- 1) Poner agua (1 litro) en una cacerola.
- 2) Encender el fuego al máximo
- 3) Colocar la cacerola al fuego y esperar que hierva el agua
- 4) Echar 2 cucharaditas de sal y dejar pasar unos segundos
- 5) Añadir lentamente la pasta, meciendo con un cucharón.
- 6) Bajar el fuego a punto medio (no mínimo que siga hirviendo si volcarse el agua)
- 7) Vigilar la pasta hasta que esté el punto de cocción (dependiendo de la pasta y del gusto de los comensales).
- 8) Apagar el fuego y sacar la pasta de la cacerola mediante un colador
- 9) Volver a colocar la pasta en un recipiente para servir.
- 10) Agregar una cucharada de mantequilla y mezclarla hasta que se derrita y distribuya.
- 11) Se puede servir con salsa o sin salsa
- 12) Agregar finalmente queso rallado



Algoritmo Mecánico

Otro ejemplo de algoritmo comúnmente empleado en los textos de enseñanza es el de los pasos para cambiar un neumático pinchado. A continuación se presenta el mismo en lenguaje natural

- 1) Estacionar el automóvil dentro de lo posible en algún lugar nivelado, encender las intermitentes, poner el freno de mano, ponerse el chaleco fluorescente y las balizas
- 2) Reunir las herramientas necesarias para el cambio (gato, llaves, llanta de refacción, etc.).
- 3) Ubicar el gato debajo del automóvil, bajo el marco de la rueda a cambiar (borde fino que recorre el auto). Si no encuentras el lugar, consulta el Manual del Propietario.
- 4) Aflojar las tuercas con la llave pertinente o de cruz, mientras aun el vehículo está totalmente apoyado en el suelo.
- 5) Levantar el vehículo utilizando el gato, hasta que el neumático a cambiar se encuentre totalmente levantado.
- 6) Seguir aflojando las tuercas hasta remover el neumático desinflado. .
- 7) Colocar el neumático de repuesto y ajustar levemente las tuercas en el orden correcto, es decir ajustando los pares de tuercas opuestas.
- 8) Bajar el vehículo
- 9) Utilizando la llave pertinente o de cruz y respetando el orden correcto, ajustar las tuercas con más fuerza.
- 10) Guardar todos los implementos y preocuparse de arreglar o inflar la rueda retirada.



Propiedades de un algoritmo

Un algoritmo presenta ciertas características que lo definen como tal y que son consideradas también propiedades fundamentales que aseguran su adecuada elaboración y su utilidad en todas las disciplinas científicas y tecnológicas. Las presentes características son adecuadas especialmente para los algoritmos en ingeniería mecánica y electrónica:

Propiedades principales

- **Abstracto:** El algoritmo es un modelo abstracto. Por lo tanto puede ser representado mediante distintos lenguajes e implementado en diferentes sistemas, mecánicos, eléctricos o electrónicos. Es decir, un algoritmo es independiente de su implementación y su representación.
- **Secuencial:** Es una sucesión o serie de pasos u operaciones ordenadas, continuas que guardan relación entre ellas.
- **Preciso:** Cada paso, operación o instrucción debe indicar de manera clara, inequívoca, precisa o legible lo que se debe realizar.
- **Finito:** tiene un límite tanto en su dimensión espacial como temporal. Debe tener un número determinado de pasos y debe producir un resultado en un tiempo definido. Un algoritmo es mejor cuanto menos pasos posea y por ende, cuanto más rápido se obtengan los resultados.

Otras propiedades

- **Funcional:** las condiciones de entrada (dominio) son datos que al procesarlos en el algoritmo (función) dan un resultado específico y único (co-domínio).
- **Determinista:** Si se prueba con las mismas entradas, debe producir los mismos resultados. No es un fenómeno aleatorio o que depende del azar. Para una definida condición inicial el algoritmo debe dar siempre el mismo resultado.

Tipos de algoritmos

Los algoritmos se pueden clasificar según su diseño, su complejidad o su función. El amplio estudio de la **algoritmia** escapa a los contenidos del presente texto, por ello solo se nombraran a modo de introducción dependiendo del interés del lector la profundización de su estudio. Algunas clasificaciones:

Según su diseño

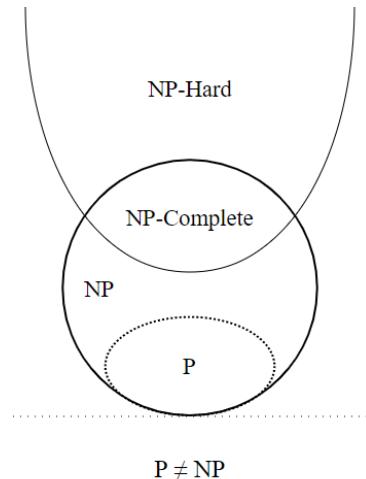
- Algoritmos determinísticos
- Algoritmos no determinísticos
- Algoritmos probabilísticos
- Algoritmos heurísticos
- Algoritmos cualitativos
- Algoritmos cuantitativos

Según su complejidad

- Clase NL
- Clase P
- Clase NP
- Clase NP-Completo
- Clase NP-Duro

Según su función

- Algoritmos de búsqueda
- Algoritmos de almacenamiento
- Algoritmos de ordenamiento
- Algoritmos de cálculo matemático
- Algoritmos de control (p.e. de sistemas mecatrónicos)
- Algoritmos genéticos, etc.



Expresión de un algoritmo

Los algoritmos se pueden expresar en dos formas principales, mediante un lenguaje informal o un lenguaje formal. En lenguaje informal es el caso del lenguaje natural, ejemplo del algoritmo mecánico, o bien mediante el lenguaje de pseudocódigo, muy aplicado entre los informáticos. En el caso de los lenguajes formales, se encuentran los lenguajes matemáticos y los lenguajes de diagramas de flujo.

Lenguajes informales

- **Lenguaje natural:**
Es el lenguaje cultural o nativo empleado por las personas en cada comunidad.
Ejemplo de este modo de expresión son el algoritmo mecánico y culinario descritos en las páginas anteriores.
- **Pseudocódigo:**
Es un lenguaje informal con cierta formalización parcial de símbolos, notación, términos y expresiones.

```
Procedimiento Ordenar (L )
//Comentario : L = (L1, L2, ..., Ln) es una lista con n elementos //
k ← 0;
Repetir
    ↑ intercambio ← Falso;
    k ← k + 1;
    Para i ← 1 Hasta n - k Con Paso 1 Hacer
        ↑ Si Li > Li+1 Entonces
            ↑ intercambiar (Li, Li+1)
            ↓ intercambio ← Verdadero;
        ↓ Fin Si
    ↓ Fin Para
    ↓ Hasta Que intercambio = Falso;
Fin Procedimiento
```

Lenguajes formales

- **Lenguajes matemáticos**

Aplicados en álgebra, cálculo superior avanzado y matemática discreta

Algoritmo de Euclides: Sean a y b enteros con $b \geq 0$ y sea $d = \text{med}(a, b)$. Entonces existen enteros m y n tales que

$$d = ma + nb.$$

DEMOSTRACIÓN. De acuerdo con el cálculo hecho antes $d = r_{k-1}$ y usando la penúltima ecuación tenemos

$$r_{k-1} = r_{k-3} - r_{k-2}q_{k-1}.$$

Así, d puede escribirse en la forma $m'r_{k-2} + n'r_{k-3}$, donde $m' = -q_{k-1}$ y $n' = 1$. Sustituyendo r_{k-2} en términos de r_{k-3} y r_{k-4} obtenemos

$$d = m'(r_{k-4} - r_{k-3}q_{k-2}) + n'r_{k-3}$$

que puede escribirse de en la forma $m''r_{k-3} + n''r_{k-4}$, con $m'' = n' - m'q_{k-2}$ y $n'' = m'$. Continuando de esta forma obtendremos una expresión para d de la forma requerida.

- **Lenguajes de diagramas**

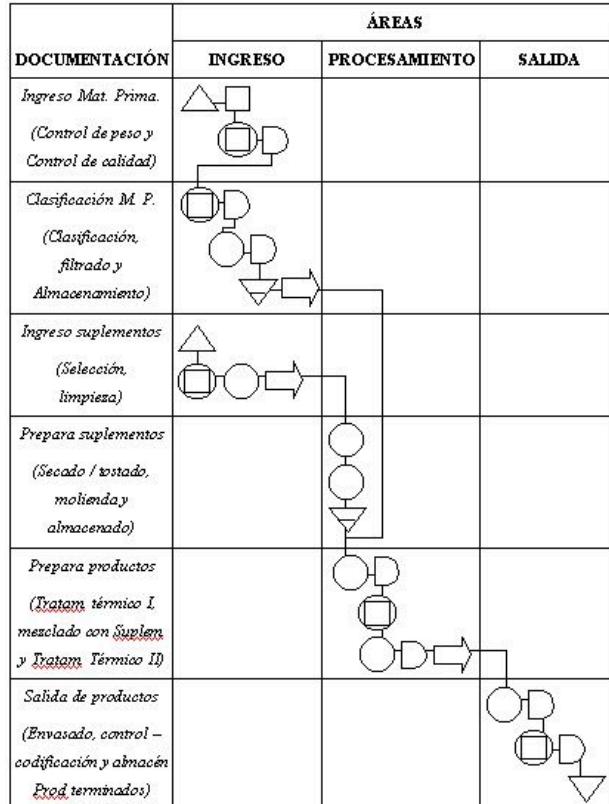
Son representaciones gráficas y hay de varios tipos según la disciplina y el algoritmo y la información a brindar. Entre los más comunes están:

- **Diagrama de flujo de procesos de producción**

Los ingenieros industriales Frank y Lillian Gilbreth, en 1921 crean el primer diagrama y que fue rápidamente adoptado por la ingeniería industrial y mecánica para la ingeniería de métodos. Y adoptado formalmente en 1940 por la Sociedad Americana de Ingenieros Mecánicos -ASME - American Society of Mechanical Engineers. Este tipo de diagramas es muy utilizado en el campo de la ingeniería industrial e ingeniería mecánica.

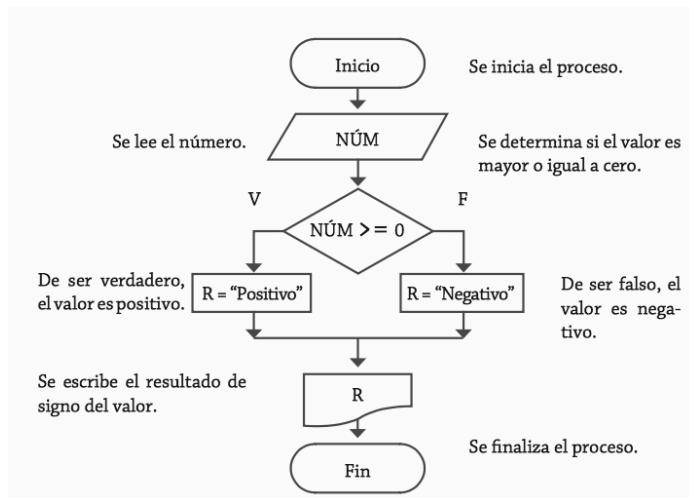
Actividad	Símbolo	Resultado Predominante
Operación	○	Se produce o se realiza algo.
Transporte	→	Se cambia de lugar o se mueve un objeto.
Inspección	□	Se verifica la calidad o la cantidad del producto.
Demora	◐	Se interfiere o se retrasa el paso siguiente.
Almacenaje	▽	Se guarda o se protege el producto o los materiales.
Actividad combinada	○□	Operación combinada con una inspección.

FLUJOGRAMA DE PRODUCCIÓN CON LA MIEL



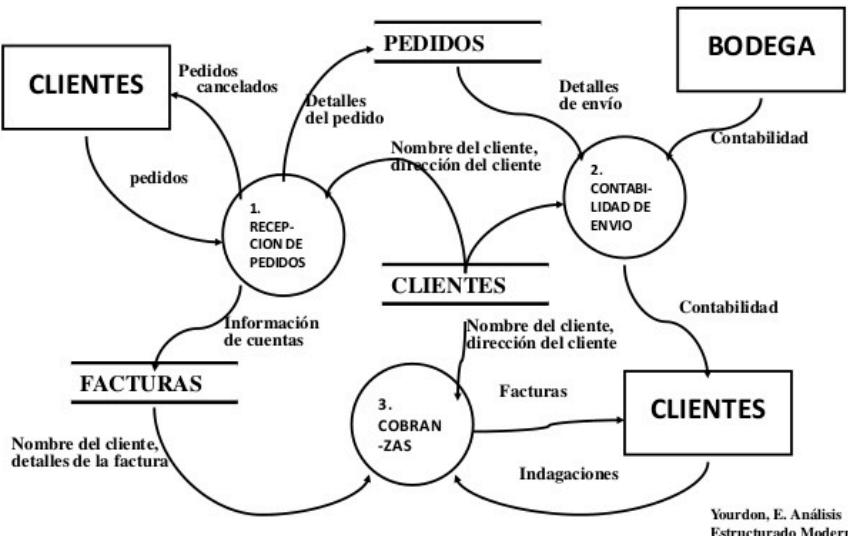
■ Diagrama de flujo de procesos electrónicos

Creado para los computadores electrónicos por *Herman Goldstine* y *John Von Neumann*, 1947, posteriormente ANSI/ISO. Adoptado principalmente por los Ingenieros electrónicos para la programación de Microcontroladores.



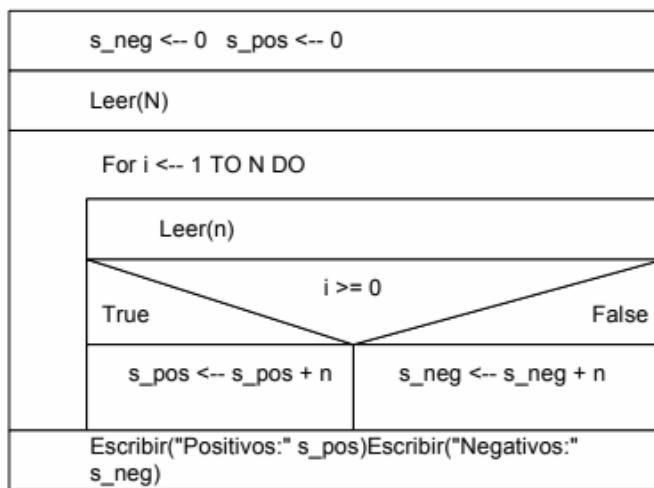
■ Diagrama de flujo de datos

Creado para el análisis de sistemas estructurado por *Ed Yourdon* y *Larry Constantine*, 1970. Este tipo de diagrama es empleado fundamentalmente en la ingeniería de sistemas.



■ Diagrama de flujo estructurado⁸

Creado por Isaac Nassi - Ben Schneiderman 1972. También conocido como Diagrama N-S. Es una evolución del diagrama de flujo de Jhon Von Neumann creado para la programación estructurada donde⁹: se *omiten las flechas de unión y los box (cajas) quedan en forma contigua..*



Implementación de un algoritmo

- Los algoritmos, como se mencionó en la definición, son construcciones conceptuales o abstractas. Son independientes de su implementación.
- La implementación de un algoritmo se puede realizar en una multiplicidad de formas. Pueden existir tantas formas de implementación como disciplinas existen: por ejemplo mecánica, eléctrica, electrónica, computación, etc. (tecnologías duras) u otras tales como recursos humanos, administración, gestión, sistemas de información, etc. (tecnologías blandas).
- Como se estudió en la reseña histórica, las primeras algoritmos fueron implementados en equipos de cómputos de construcción mecánica (con engranajes), posteriormente aparecieron los electromecánicos (con relevadores o relés) y luego los electrónicos (initialmente a válvulas electrónicas, luego transistores y hoy circuitos integrados). Incluso actualmente se encuentra en etapa de experimentación (científica/no comercial) computadoras cuánticas.
- Actualmente, los algoritmos para computadoras digitales se implementan mediante los lenguajes de programación: C, C++, C#, Java, Python, etc.

⁸ También conocidos como Diagrama de Chapin por el Dr. Ned Chapin.

(<https://www.legacy.com/obituaries/sfgate/obituary.aspx?n=ned-chapin&pid=174013661>)

⁹ Fundamentos de informática. Facultad Regional de Rosario. Universidad Tecnológica Nacional.

https://www.fro.utn.edu.ar/repositorio/catedras/electrica/2_anio/fundamentos_informatica/apuntes/Introd_Program.pdf

Desarrollador de algoritmos¹⁰

Un desarrollador (developer) de algoritmos, tiene en cuenta determinados pasos para su creación, pasos que se utilizan en todos los emprendimientos tecnológicos. Estos se estructuran en 5 etapas fundamentales a saber, los cuales no se profundizará, pues se desarrollarán en otras asignaturas de la carrera

- **Etapa 1: Identificación**

Es la etapa de identificación y definición del problema. En tecnología hay un dicho: Un problema bien planteado es un problema resuelto. Algunos autores definen esta etapa como **identificación de la oportunidad**, o **la necesidad**. Esto se debe a que el ingeniero es capaz de comprender las situaciones de la realidad y dar su apropiada solución al problema desde la **inversión de nuevas tecnologías**. Esta etapa se visualiza y plantea desde la necesidad del cliente. Las computadoras se hicieron para resolver problemas del mundo y por ello la informática (como tecnología aplicada) pone su mirada en el contexto:

- **Etapa 2: Análisis**

Etapa en la cual hay que profundizar sobre la necesidad o el problema del cliente. Por ello es una etapa de investigación y exploración, para recabar la mayor información posible y poder entender el contexto y analizar la situación en la cual se está trabajando. Es importante recurrir a todas las fuentes posibles y determinar si hubo antecedentes al respecto y Determinar los recursos necesarios en pos de la solución

- **Etapa 3: Diseño**

Se generan ideas, bosquejos, alternativas. En nuestro caso se diseña apropiadamente el algoritmo en forma eficiente, óptima para resolver un problema desde un punto de vista conceptual expresándose en el lenguaje adecuado en función del sistema analizado

- **Etapa 4: Implementación**

Es la etapa de la planificación, gestión e implementación. Este tema se desarrollará en la **unidad 2**. Donde se implementarán los algoritmos con un lenguaje de computación digital..

- **Etapa 5: Evaluación**

Etapa transversal a todas las etapas anteriores. Todas requieren evaluarse para optimizar el proceso. (trabajo realizado bajo normas de calidad que permiten gestionar y documentar apropiadamente cada etapa)

¹⁰

<https://www.universidadviu.com/int/actualidad/nuestros-expertos/que-es-el-proceso-tecnologico-y-cuales-son-sus-fases>

PSEUDOCÓDIGO

Definición

El pseudocódigo es un falso código. **Pseudo-** es un prefijo que significa falso (actualmente aceptado **seudo**¹¹). Es el lenguaje informal empleado comúnmente por los informáticos para comunicar, analizar y corregir sus proyectos. El pseudocódigo es una de las formas de expresar un algoritmo en la **etapa diseño**.

Sintaxis

¿Cómo se escribe un algoritmo en pseudocódigo?. Un pseudocódigo se conforma de una **parte formal** o normalizada (a través de un acuerdo de los integrantes de la comunidad científica o tecnológica que lo emplean) y una **parte informal** que se utiliza con libertad siempre y cuando cumpla con las propiedades de un algoritmo: preciso y conciso. El pseudo-código se lee igual que el lenguaje natural, de arriba hacia abajo y de izquierda a derecha.

Parte formal

Esta parte se encuentra conformada por un conjunto palabras reservadas: **instrucciones**, **variables**, **símbolos** para las operaciones fundamentales y finalmente la **arquitectura** del pseudocódigo, es decir como estará estructurado.

- Las **palabras reservadas** son palabras del lenguaje natural que se apartan para darles una función especial dentro del pseudocódigo. Estas palabras reservadas y apartadas se denominan instrucciones. Si se requiere emplear una palabra reservada para definir o realizar otra operación se deberá recurrir a un sinónimo. Ejemplo de palabras reservadas en el pseudocódigo son: **algoritmo**, **comienzo**, **final**, **leer** (para introducir datos) y **mostrar** (para mostrar datos)
- Los **comentarios**, son una parte del algoritmo que sirve para explicar y recordar posteriormente la simbología expresada. Para escribir un comentario de una sola línea se realiza comenzando con dos barras, por ejemplo: **//mi algoritmo**. Para escribir un comentario de varias líneas o renglones se realiza entre los símbolos **/* y */**. Por ejemplo **/* este es un comentario muy largo que tiene varios renglones o líneas y debo escribirlo entre los símbolos de asterisco y barra diagonal */**.
- Las **variables**, deben siempre empezar con una letra puede ser mayúscula o minúscula. Pueden contener mayúsculas, minúsculas, números y el símbolo de guión bajo (_). Por ejemplo: **Numero1**, **Numero 2**, **Pers Nombre**, **Pers Apellido**, **Pers_DNI** son variables

¹¹ <https://dle.rae.es/seudo>

correctamente definidas. Y **Número1**, **2numero**, **_numero** son ejemplos de variables mal definidas. Normalmente no se emplean acentos ya que las letras con acentos corresponden a caracteres especiales en ASCII y pueden generar complicaciones en la implementación posterior del algoritmo. Sin embargo, como se trata de un pseudocódigo en lenguaje natural, se puede permitir colocar las variables y expresiones con acento y diéresis recordando que en su posterior implementación se deberán descartar.

- Las **operaciones aritméticas fundamentales** son multiplicación (*, x), división (/), resta (-), suma (+), y se agregan dos muy útiles que son módulo (%, mod) el resto de una división y potencia (^, ↑).
- Las **operaciones lógicas fundamentales**, conjunción (AND, &, &&), disyunción (OR, ||), negación (NOT, ~, !), igual (=, ==), diferente (≠, !=, <>).
- La **estructura del algoritmo** habitual y acordada es: un bloque principal que se inicia con la palabra reservada inicial de **algoritmo** seguida de un nombre con formato de variable que define el algoritmo en forma general. Luego posee un bloque principal que comienza con la palabra reservada **Comienzo** y se finaliza con la palabra reservada **Final**. Las definiciones y declaraciones de variables se realizan al inicio del algoritmo, continuado por la lectura o introducción de datos, luego el proceso de cálculos y finalmente la salida o impresión de resultados obtenidos.

Para que la estructura tenga una mejor visibilidad, los algoritmos se pueden escribir con birome de varios colores. Por ejemplo: los comentarios en **color verde**, las palabras reservadas en **negro** y el resto del contenido en **azul** incluyendo las variables.

Parte informal

La parte informal, es todo aquel contenido que no se ha formalizado, es decir que no se ha acordado su uso y no se ha definido en la comunidad científica-tecnológica. Por lo tanto se puede emplear libremente la parte informal siempre y cuando se respete las características de un algoritmo preciso y conciso y normalmente se escribe en color azul.

Ejemplo

El presente pseudocódigo es un ejemplo de algoritmo diseñado para calcular el área de un círculo. A continuación se explica su estructura.

Primero se define el **título** del algoritmo que comienza con la palabra reservada **algoritmo** seguido de un **nombre** que debe tener la estructura de una **variable** (*Algoritmo Area_Circulo*).

Luego hay un **comentario** de varias líneas que permite explicar en detalle de que se trata dicho algoritmo (*/* ... texto ... */*).

Con la palabra reservada **Comienzo** se da inicio al bloque principal del algoritmo.

Se **definen las variables** (las variables pueden ser de varios tipos: *enteras, reales, palabras*).

Se indica un **comentario** de una línea (// texto.), expresando que la siguiente **instrucción** corresponde a la lectura de datos.

Luego se **lee** el dato. Para ello se realiza la **instrucción** con la palabra reservada **leer** acompañada de la **variable** donde se guardará el dato (*Leer Radio*).

Entramos a la parte de **procesamiento**, Se realizan los cálculos (la computación) a través de las operaciones aritméticas necesarias.. ($PI \leftarrow 3,14$, $Area \leftarrow PI * Radio ^ 2$)

Finalmente se muestra el **resultado** por pantalla (*Mostrar "El área del círculo es:", Área*), lo que está entre comillas se imprime textual y lo que no está entre comillas es una variable y se imprime su valor.

```
Algoritmo Area_Circulo
/*
    El presente algoritmo matemático realiza
    cálculos del área de una figura geométrica
*/
Comienzo

    // Definición de variables y constantes
    Definir Pi, Radio, Area como Real;
    Definir Perimetro, Superficie como Real;

    // Introducir como dato el Radio del círculo.
    Leer Radio;

    // Cálculo del área
    PI ← 3.14;
    Area ← PI * Radio ^ 2;

    // Mostrar el resultado
    Mostrar "El área del círculo es:", Area;

Final
```

DIAGRAMA DE FLUJO

Definición de diagrama de flujo

El diagrama de flujo, es la representación gráfica de un algoritmo, en el cual se realiza una esquematización de los procesos organizándose visualmente en forma secuencial.

El diagrama para procesamiento electrónico

De todo los tipos de diagrama de flujo que existen para procesar los distintos tipos de datos e información en las múltiples disciplinas de ingeniería, emplearemos específicamente el **diagrama de flujo para el procesamiento electrónico de datos** ideado por Von Neumann en 1947 para las computadoras digitales. Posteriormente fue adoptado por el Dr. Ned Chapin el cual denominó **diagrama de flujo para el procesamiento de información**.

Estandarización ANSI

El diagrama de flujo es un **lenguaje formal** ya que su implementación fue normalizada por la organización ANSI¹² para procesos electrónicos alrededor del 1960¹³. Como, el **software** se independizó definitivamente del **hardware** y su desarrollo evolucionó vertiginosamente, el mismo diagrama adoptado por las normas ISO/IEC¹⁴ fue denominado: **diagrama de flujo para el procesamiento de datos** o procesamiento de la información¹⁵.

Simbología para la diagramación

- Cada símbolo está asociado a una función específica y no puede emplearse para otra función.
- Los **símbolos elementales** son empleados en todos los procesos básicos mientras que los **símbolos especializados** son utilizados para realizar procesos más complejos y detallados.
- Algunos símbolos, que podrán encontrar en la bibliografía de documentos originales del momento, quedaron obsoletos debido a que fueron creados para el hardware de esa época, especialmente aquellos relacionados con ingreso de datos, salida de información o almacenamiento. Ejemplo de estos son los símbolos empleados para tarjetas perforadas, cinta magnética, etc.

¹² **ANSI:** American National Standards Institute, Instituto Nacional de Normalización de EEUU.

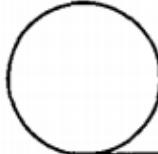
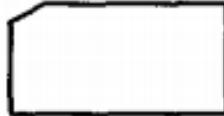
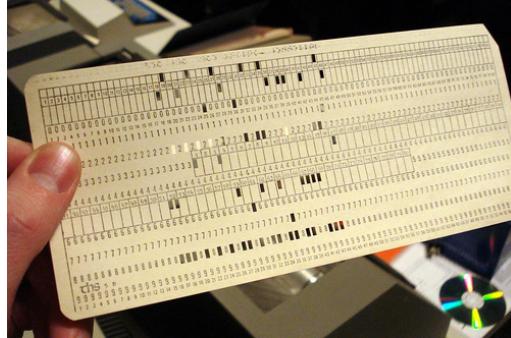
¹³ Publicación de la normalización ANSI del diagrama de flujo

http://gr.xjtu.edu.cn/c/document_library/get_file?p_l_id=586332&folderId=583398&name=DLFE-9604.pdf

¹⁴ **ISO:** International Organization for Standardization, Organización internacional para la estandarización. **IEC:** International Electrotechnical Commission, Comisión internacional de Electrotecnia.

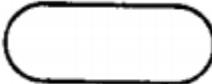
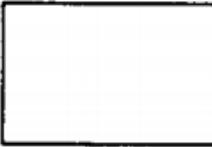
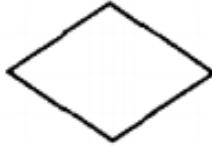
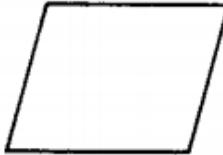
¹⁵ https://webstore.ansi.org/preview-pages/ISO/preview_ISO+5807-1985.pdf

Símbolos obsoletos

Nombre	Símbolo	Hardware
Cinta Magnética (Magnetic Tape)		
Tarjeta Perforada (Punched Card)		
Cinta Perforada (Punched Tape)		
Monitor CRT (Tubo de Rayos Catódicos) (Display)		

Aunque estos símbolos no son comúnmente utilizados para fines comerciales o industriales, pueden encontrarse en algunos casos específicos.

Símbolos fundamentales

Nombre de proceso	Nombre de la figura	Símbolo	Descripción
Terminal (inicio / final)	Ovalo		Se emplea para realizar el inicio o finalización de un algoritmo. Por lo tanto es el primer símbolo empleado y el último en dibujarse. Por ello su nombre de terminal.
Procesamiento (operaciones aritméticas)	Rectángulo		Símbolo empleado para los procesos. Empleado para hacer la asignación de datos a variables o hacer los cálculos y operaciones necesarias en el algoritmo. Fundamentalmente son operaciones de tipo aritmética.
Decisión (operaciones lógicas)	Rombo		Punto donde se decide entre SÍ o NO se sigue una secuencia determinada o define una condición si es V (verdadera) o F (falsa). Fundamentalmente son operaciones de tipo lógica.
Comunicación (Entrada/Salida)	Paralelogramo		Es el símbolo que indica la comunicación del sistema con su contexto (Interface). En la simbología o procesamiento básico se utiliza el mismo símbolo para el ingreso de datos y la salida de información. La simbología, como se puede observar, es independiente de las características de hardware por la cual se realice la entrada de datos o salida de información. Tampoco indica si se realiza en forma manual o automática.
Líneas de flujo	Flecha		Son las líneas que indican el sentido y dirección del flujo de datos e información secuencial.

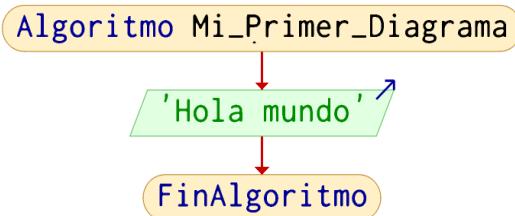
Símbolos especializados

Nombre de proceso	Símbolo	Descripción
Comentario		Anotaciones. Comentarios. Descripción. Cuando los algoritmos son muy complejos y extensos, se requiere aclarar o describir los procesos.
Almacenamiento Central		Actualmente, este símbolo hace referencia al almacenamiento interno, lo que se conoce como memoria RAM (Memoria de acceso aleatorio)
Procesamiento Predefinido		Son las llamadas subrutinas o funciones. Son algoritmos dentro de otro algoritmo que realizan funciones predefinidas, por ello también se llaman funciones.
Conector fuera de página		Cuando el algoritmo es muy largo y se requiere continuar el diseño en otra página (hoja)
Conector en la página		Conector dentro de una misma página. Cuando el diagrama es complejo y decide continuar parte del código en otro sector de la misma hoja.
Almacenamiento de disco		Almacenamiento en disco o base de datos. Es un símbolo antiguo pero que evolucionó en su concepto
Salida por pantalla		El nuevo símbolo de visualización. Puede ser un display, un monitor, una pantalla plana o tubo de rayos catódicos, etc.
Ingreso de datos manualmente		Son datos que se ingresan por teclado, por lector de código, etc. que requieren de la intervención de una persona
Documento Medio de salida		Impresión de un documento para su visualización. Por ejemplo en una impresora.

No se explican todos los símbolos avanzados o especializados, sólo aquellos que pueden llegar a utilizarse durante la cursada y son aplicables a los procesamientos electrónicos en los cuales nos focalizamos para dar fundamento a los contenidos que se verán el año siguiente en informática industrial.

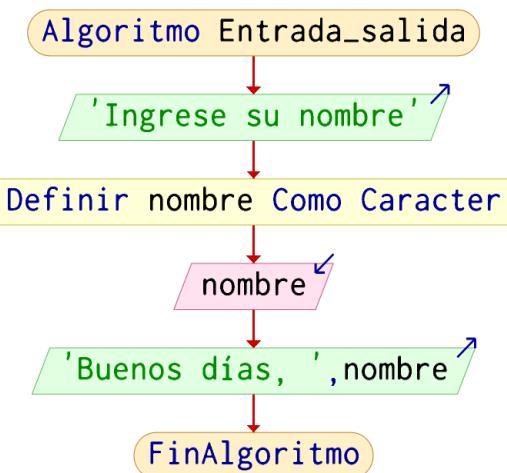
Ejemplos de diagramas de flujo

Ejemplo 1: Mi primer diagrama



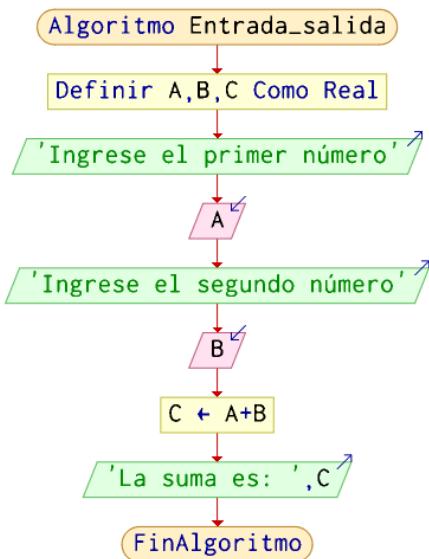
- Diagrama simple con solo tres figuras.
- Los dos diagramas de forma de ÓVALO, dan inicio y fin al diagrama de flujo, son terminales.
- El diagrama intermedio en forma de PARALELOGRAMO es una interfaz de salida. Muestra desde el sistema digital un contenido alfanumérico, un mensaje: "HOLA MUDO".
- En este diagrama, no se indica si la salida es por monitor (pantalla), impresora (papel) o altavoces (audio). No indica el tipo de periférico de salida. Solo indica que se emite un mensaje desde el sistema al usuario..

Ejemplo 2: Entrada y salida



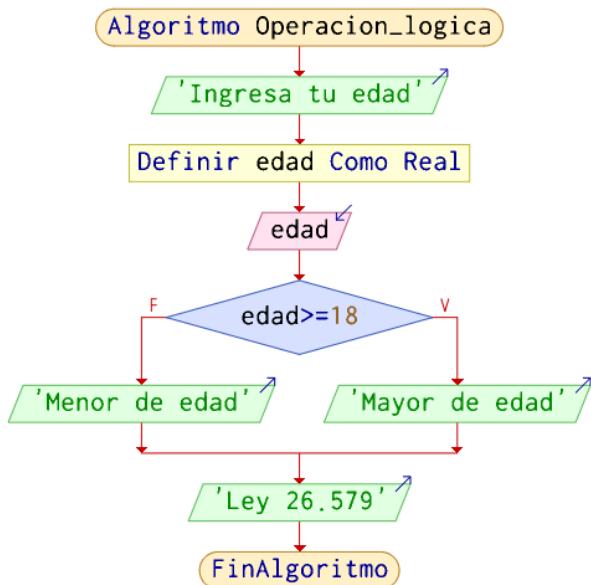
- Los ÓVALOS indican el inicio y el final del algoritmo.
- El primer PARALELOGRAMO es un mensaje del sistema al usuario.
- El segundo diagrama, un RECTÁNGULO, es un proceso que define el nombre y el tipo de variable.
- El segundo PARALELOGRAMO, es un input, es una entrada de datos desde el usuario al sistema. Lo que ingrese el usuario se guardará en la variable nombre. El input, se puede realizar por cualquier periférico de entrada, por ejemplo un teclado, un escáner, un micrófono, etc.
- El tercer PARALELOGRAMO de salida muestra dos textos, uno predefinido, fijo, escrito entre comillas "buenos días" y el otro variables según el datos ingresado en nombre. La primera coma es parte del texto fijo, en cambio la segunda coma es parte del lenguaje y sirve para separar entre la cadena de caracteres y la variable.

Ejemplo 3: Sumar dos números



- Ejemplo de operaciones aritméticas.
- En el primer RECTÁNGULO se define el nombre y tipo de variables.
- El primer PARALELOGRAMO muestra un mensaje, indica al usuario que ingrese un número.
- El segundo PARALELOGRAMO lee el número ingresado al sistema por el usuario y lo guarda en la variable A.
- El tercer y cuarto PARALELOGRAMO se repite la operación pero con la variable B.
- El RECTÁNGULO es un proceso. Realiza la operación aritmética de suma entre los valores de las variables de A y B y el resultado lo ("guarda") asigna a la variable C.
- En el quinto y último PARALELOGRAMO se muestra el mensaje encerrado entre comillas seguido por el valor de la variable C.

Ejemplo 4: Verdadero ó Falso



- El primer PARALELOGRAMO emite del sistema al usuario un mensaje. Le dice que ingrese su edad.
- En el RECTÁNGULO se define el nombre y tipo de la variable a emplear.
- En el segundo PARALELOGRAMO, el sistema lee del periférico de entrada la edad ingresada por el usuario y la "guarda" (asigna) en la variable edad..
- El ROMBO, realiza la operación lógica sobre la edad y toma una decisión según la proposición establecida.
- Proposición: Es mayor o igual a 18 años. Si es así, entonces la proposición es verdadera y seguirá el camino indicado con la letra V y mostrará el mensaje de Mayor de edad. En caso contrario seguirá el camino indicado con la letra F.
- El último mensaje: Ley 26.579. Independientemente del camino que recorra V o F, siempre se emitirá.