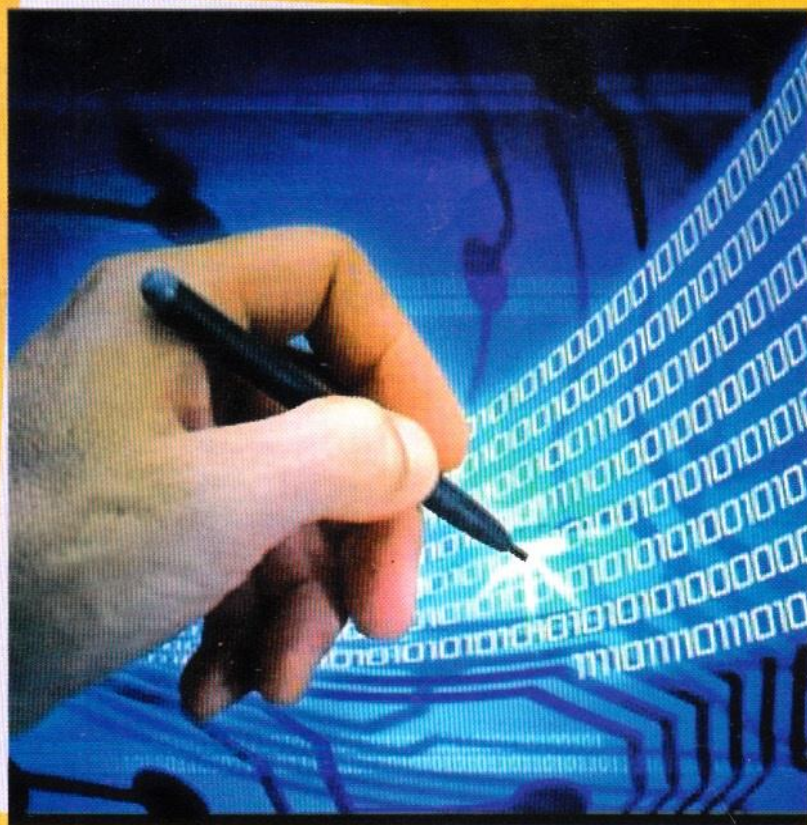


EL ARTE PARA PROGRAMAR COMPUTADORAS



CARLOS RODRÍGUEZ GRÁNDEZ



Esta obra está bajo una [Licencia Creative Commons Atribución- NoComercial-Compartirigual 2.5 Perú](#).

Vea una copia de esta licencia en <http://creativecommons.org/licenses/by-nc-sa/2.5/pe/>



El arte para programar computadoras

Carlos Rodríguez Grández

El arte para programar computadoras

El arte para programar computadoras

Autor - Editor

© Carlos Rodríguez Grández

Jr. Prolongación Tacna N° 175

Huayco – Tarapoto

Primera edición, Abril de 2019.

Tiraje 1000 ejemplares

Hecho el depósito legal en la Biblioteca Nacional del Perú N°
2019-03050

ISBN N° 978-612-00-4276-2

Impreso en:

Texcom

De: Carlos Alberto Uriarte Quevedo

Jr. Ricardo Palma 321

San Martín - Tarapoto

Presentación

La génesis de este libro se remonta a diciembre del año 2017, motivado por los alumnos y la necesidad de fortalecer los conocimientos base para convertirse en un artista algorítmico. Esa creatividad innata en las personas, indistintamente del nivel educativo en que se encuentre, y siendo la computadora una herramienta fundamental en nuestra vida diaria, agente de cambio y mejora continua, me ha inspirado denominar el libro como “El arte para programar computadoras”, cuyo contenido principal presenta algoritmos usando diagrama de flujo de datos y el lenguaje de programación *Python*, organizado en capítulos que respetan una secuencia lógica y encaminan al lector al desarrollo de sus competencias.

Para la construcción del contenido se tomó en cuenta las recomendaciones y sugerencias de profesionales expertos en programación de computadoras y catedráticos universitarios.

Tarapoto, abril del 2019.

El autor.

Prólogo

A través del arte podemos encontrar nuestra humanidad y a nuestro Creador, esta actividad requiere generalmente tiempo, dedicación y paciencia, sobre todo paciencia. Si el arte trasciende nuestra existencia se agrega un factor elemental que ha llevado al hombre a considerarse un ser excepcional, este factor es la pasión.

Pasión que se manifiesta en esta obra: “El arte para programar computadoras”, donde de manera fresca y elocuente se detalla los conceptos necesarios que llevan a desarrollar la lógica de un programador capaz de aplicar sus conocimientos en cualquier parte del mundo, ya que su carácter universal engloba lo necesario para desenvolverse como un agente transformador de las tecnologías de la información y comunicaciones en nuestra sociedad.

Recomiendo la lectura y reflexión de este libro, el cual permitirá motivar, desarrollar y afianzar tus conocimientos sobre el arte para programar.

Cristian Werner García Estrella

Prefacio y agradecimiento

El presente libro, “El arte para programar computadoras”, cumple con uno de los deseos logrados es proporcionar conocimientos base para ser creativo y hacer de la computadora una herramienta fundamental en nuestra vida diaria. He plasmado algoritmos utilizando el diagrama de flujo de datos y el lenguaje de programación *Python*, organizando en capítulos que respeten una secuencia lógica; y encaminar al lector al desarrollo de sus competencias. Y ha sido posible, gracias al apoyo incondicional de muchas personas, a quienes va mi agradecimiento y les deseo muchos éxitos.

A Dios, por darme la vida y una familia maravillosa.

A la Universidad Nacional de San Martín – Tarapoto, Facultad de Ingeniería de Sistemas e Informática, por brindarme las facilidades durante la licencia por año sabático, en especial a los Ingenieros catedráticos John Clark Santa María Pinedo y Richard Enrique Injante Oré, quienes asumieron mi carga académica, año 2018; y al Ingeniero catedrático Cristian Werner García Estrella, quién prologa este texto; todos ellos alentaron siempre con entusiasmo nuestro trabajo.

Índice

Presentación	5
Prólogo	6
Prefacio y agradecimiento	7
Introducción	11
Lista de términos	12
Dedicatoria	13
CAPÍTULO I	
ARTE Y RECURSOS COMPUTACIONALES	14
1.1. El arte	14
1.2. Recursos Computacionales	15
1.2.1. El procesador	15
1.2.2. El almacenamiento	17
1.2.3. Los dispositivos de entrada y salida	18
1.2.4. Los buses de conexión	20
Preguntas de repaso	23
Recapitulación del arte	25
CAPÍTULO II	
ANÁLISIS DEL ENUNCIADO	26
2.1. Enunciado de un problema	27
2.2. Análisis del enunciado	30
Preguntas de repaso	32
Recapitulación del arte	33

CAPÍTULO III

DATOS IDENTIFICADOS PARA EL ARTE	34
3.1. Datos para el arte	34
3.2. Identificadores	35
3.3. Palabras reservadas	35
3.4. Tipos de datos	36
3.5. Variables	38
3.6. Constantes	39
3.7. Contadores	39
3.8. Acumuladores	39
3.9. Operadores	39
3.10. Expresión	42
3.11. Ejemplos	42
Preguntas de repaso	49
Recapitulación del arte	52

CAPÍTULO IV

TÉCNICA PARA EL ARTE	53
4.1. Técnica de diagrama de flujo de datos	53
4.2. Símbolos en el DFD	54
4.3. Construcción de los DFD	59
Preguntas de repaso	61
Recapitulación del arte	64

CAPÍTULO V

DISEÑO DEL ARTE	65
5.1. Estructura secuencial	65
5.2. Estructura de control de decisión simple	68
5.3. Estructura de control de decisión múltiple	72
5.4. Repeticiones	76
5.4.1. Estructura PARA	76
5.4.2. Estructura MIENTRAS	78
5.4.3. Estructura HACER MIENTRAS	82
Preguntas de repaso	85
Recapitulación del arte	99

CAPÍTULO VI

PROGRAMACIÓN DEL ARTE	100
6.1. El lenguaje de programación Python	101
6.2. Entorno de desarrollo del lenguaje de programación Python	103
6.3. Sintaxis básica del lenguaje de programación Python	108
Preguntas de repaso	126
Recapitulación del arte	126
Anexos	127
Anexo 1. Otros recursos sobre Python	128
Glosario de términos	129
REFERENCIAS BIBLIOGRÁFICAS	130

Introducción

El propósito del presente libro es que el lector adquiriera conocimientos base, para convertirse en un algoriólogo. Para lo cual, requiere ser creativo. Esta actividad creativa es lo que llamaremos el arte para programar computadoras.

Este libro está organizado en seis (6) capítulos en forma secuencial, sin condicionar la lectura; pudiendo el lector dirigirse al capítulo de su interés. Cada capítulo comienza con una frase célebre, la cual sirve para pretexto de reflexión del lector. En el capítulo I encontrará una explicación del arte (algoritmo) con los recursos físicos de cualquier computador. El capítulo II contiene todo lo relacionado al análisis del enunciado planteado o problema a solucionar. En el capítulo III se idéntica los datos encontrados en el enunciado. El capítulo IV contiene la técnica usado en este libro, en el capítulo V está el diseño del arte hasta las estructuras repeticiones y en capítulo VI está la programación del arte, usando el lenguaje de programación *Python*.

En cada capítulo se incluyen preguntas de repaso para que el alumno pueda determinar si necesita estudiar más el tema o puede seguir adelante. Al final se hace la recapitulación del arte que es un resumen conciso, organizado en función de los principales temas abordados.

Lista de términos

- ATA.-** Accesorio de tecnología avanzada en inglés *Advanced Technology Attachment*
- BIOS.-** Sistema básico de entrada y salida en inglés *Basic Input/Output System.*
- CPU.-** Unidad central de procesos en inglés *Central Processing Unit.*
- DFD.-** Diagrama de flujo de datos.
- DNI.-** Documento nacional de identidad.
- DRAM.-** RAM dinámica.
- IDE.-** Entorno integrado de desarrollo en inglés *Integrated Development Environment.*
- IDLE.-** Entorno integrado de desarrollo y aprendizaje en inglés *Integrated Development Environment for Python.*
- IGV.-** Impuesto general a las ventas.
- RAM.-** Memoria de acceso aleatorio y volátil en inglés *Random Access Memory.*
- ROM.-** Memoria de solo lectura y no volátil en inglés *Read Only Memory.*
- SATA.-** Serial ATA.
- SRAM.-** RAM estática.

Dedicatoria

A mi esposa y mis hijos, por el apoyo incondicional en todos mis proyectos de vida.

A mis padres por inculcarme valores morales.

CAPÍTULO I

ARTE Y RECURSOS COMPUTACIONALES

La ciencia es lo que entendemos lo suficientemente bien como para explicarlo a una computadora; el arte es todo lo demás.

Donald E. Knuth

1.1. El arte

Según Guillermo M. (2018), el arte es un algoritmo.

Según Rosenblatt D. (2018), nos reafirma que, desde su invención, la computadora ha sido considerada un agente de racionalización y perfeccionamiento. Ella representa la lógica y la eficiencia, generalmente lo opuesto a la imaginación y sensibilidad necesarias para la creación artística; sin embargo, la tormenta parece acercarse insidiosa a esta última barrera de la humanidad. La creatividad ha representado siempre una de las características únicas en el proceso evolutivo de nuestra especie, por lo tanto, no es extraño que la posibilidad de ser forzados fuera de este campo por una computadora nos cause un complejo de inferioridad inmediato.

1.2. Recursos Computacionales

Tomando en cuenta la versatilidad de las computadoras en todos los campos de la actividad humana y antes de inmiscuirse en la programación, presentamos en la figura 1 las partes básicas de los recursos computacionales, aquellos elementos físicos (hardware) que conforman una computadora, agrupados en:

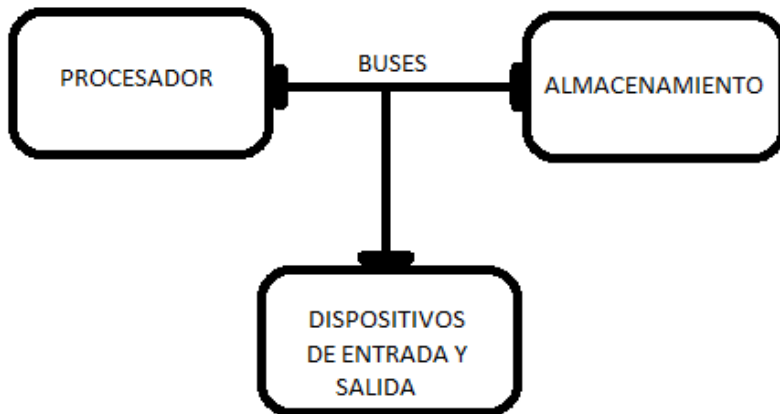


Figura 1. Recursos computacionales básicas

1.2.1. El procesador

El procesador, es la unidad central de procesos. Se encarga de ejecutar las instrucciones de las aplicaciones que le solicita el sistema operativo y controla el tratamiento de los datos.

El CPU es una especie de cerebro del computador, y físicamente contiene 3 partes: la unidad aritmética lógica, la unidad de control y los registros.

Es importante indicar que actualmente los “núcleos” o “cores” de un CPU, son los procesadores que están empaquetados en un único chip, y que, realizan su tarea por separado. Ahora podemos hablar de tecnología de CPU multinúcleo, para realizar varias tareas a la vez. Es el sistema operativo que se encarga de hacer sentir al usuario como si se realiza en paralelo las acciones del computador.

La tecnología de CPU multinúcleo ha sido posible gracias al campo de la miniaturización (componentes electrónicos en un espacio muy pequeño), y los fabricantes presentan diferentes gamas de acuerdo al nivel de prestación que ofrece cada uno.

En la figura 2 se muestra lo que ofrecen algunas marcas:

Intel ofrece Celeron, Pentium, Core i3, Core i5, Core i7 y Core i9.

AMD ofrece Sempron, Athlon, APUs, FX 4300, FX 6300, FX 8300, RYZEN 5, RYZEN 3 y RYZEN Threadripper.

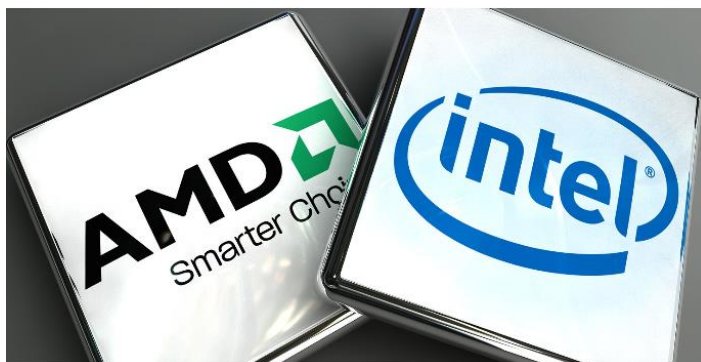


Figura 2. Algunas marcas de CPU

1.2.2. El almacenamiento

Hablar de almacenamiento o memoria es referirnos a la forma de organizar los datos dentro del computador. En cuanto al tipo de almacenamiento, en la figura 3 observamos, el almacenamiento a largo plazo (Memoria ROM) y almacenamiento a corto plazo (Memoria RAM). Ambos son indispensables para el funcionamiento del computador.

Memoria ROM, lugar donde se guardan los datos a utilizar cuando sean requeridos por el computador. Aquí se encuentran los denominados Discos Duros y los BIOS que contiene información básica para que cada computador arranque.

Existen Discos Duros internos (instalados en la tarjeta madre del computador) y Discos Duros externos (transportables con diversidad de presentaciones y fácil conexión al computador).

Ambos disponibles con capacidad de almacenamiento de varios terabytes al gusto del cliente.

Memoria RAM, lugar donde se guardan las instrucciones y los datos mientras son utilizados por el computador. Es la memoria principal del computador. Existen dos tipos básicos de memoria RAM: DRAM y SRAM.



Figura 3. Tipos de Memorias

1.2.3. Los dispositivos de entrada y salida

Estos dispositivos permiten la entrada y salida de datos del computador mediante diferentes métodos. Establecen y efectúan la comunicación entre el computador y el usuario. Existen varios tipos de dispositivos, como observamos en la figura 4. Algunos se comportan como dispositivos de entrada y salida, a estos dispositivos se les denomina dispositivos mixtos.



Figura 4. Tipos de dispositivos de entrada y salida

Lista de dispositivos de entrada:

Mouse, transmite los movimientos de nuestra mano con el computador.

Teclado, transmite instrucciones, textos y números entre el usuario y el computador.

Scanner, capta una imagen y lo conduce mediante espejo de luz hacia el computador.

Webcam, es una cámara que está conectado al computador.

Lápiz óptico.

Joystick.

Lista de dispositivos de salida:

Monitor o pantalla.

Impresoras.

Plóters.

Bocinas.

Altavoces.

Auriculares.

Fax.

Lista de dispositivos mixtos:

USB.

Cámara fotográfica.

1.2.4. Los buses de conexión

Los buses sirven para que los recursos computacionales se comuniquen. Físicamente (*hardware*) los buses son cables que semejan a una autopista de datos. Cada cable de bus se llama PIN. El número de PIN depende del número de bytes que pueden enviar a la vez. En la figura 5 se muestra un bus de conexión donde la información viaja en forma de 0 y 1 (bits).

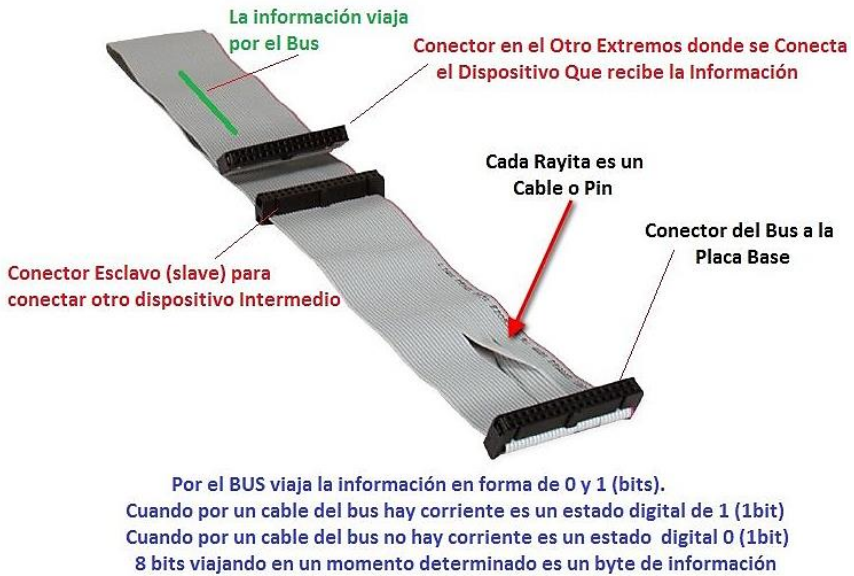


Figura 5. Bus de conexión

Fuente: <http://www.areatecnologia.com/informatica/tipos-de-buses.html>

Los buses de conexión por su ubicación se clasifican en: buses internos (transportan los datos dentro de la placa madre) y buses externos (transportan los datos a los dispositivos de entrada y salida) al computador. En la figura 6 se muestra un bus de conexión interna.

Los buses de conexión por su función se clasifican en ATA y SATA.

En cada cable de bus encontraremos tres tipos de buses:

- Bus de datos, transporta los datos en forma unidireccional.
- Bus de direcciones, transporta la ubicación a donde llevar los datos en forma unidireccional.
- Bus de control, transporta los comandos y señales del estado de los recursos computacionales.

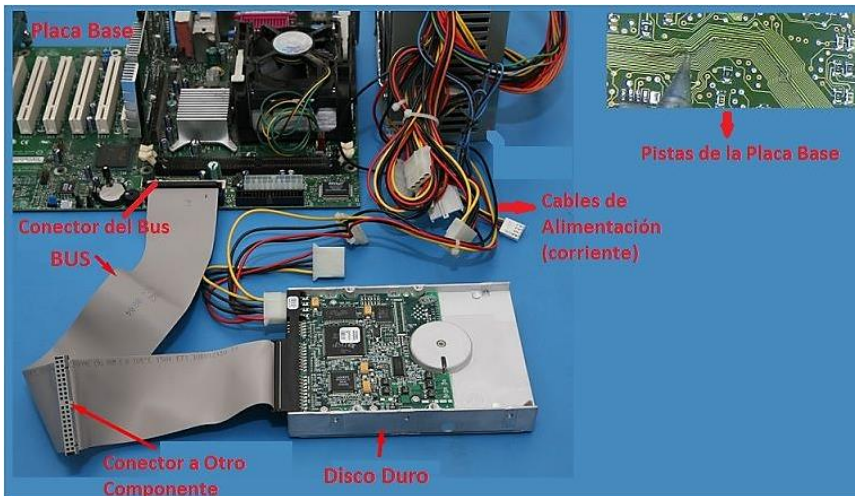


Figura 6. Bus de conexión interna

Fuente: <http://www.areatecnologia.com/informatica/tipos-de-buses.html>

Preguntas de repaso

Marcar verdadero (V) o falso (F)

1. () La Unidad Aritmética Lógica procesa los datos recibidos y envía los registros.
2. () La memoria RAM es la memoria inmediata que utiliza el sistema operativo para almacenar temporalmente la información y programas con que el usuario trabaja.
3. () Son elementos físicos de un computador: el procesador, los dispositivos de entrada y salida, los dispositivos de almacenamiento y los buses.
4. () Los buses (Bus de Datos, Bus de Direcciones y Bus de Control) tienen la tarea de interconectar los elementos del computador.
5. () El procesador, es la unidad central de procesos que se encarga de ejecutar las instrucciones que le da el sistema operativo.
6. () A través de los buses de direcciones circulan las señales de control de todo el sistema.
7. () La memoria ROM almacena de forma masiva los datos a largo plazo.
8. () La Unidad de Control analiza los resultados obtenidos y los envía a otros dispositivos.
9. () El teclado, el mouse, los parlantes, la Impresora, una USB son dispositivos de entrada y salida.

10. () El arte es un algoritmo.
11. () La computadora es considerada un agente de racionalización y perfeccionamiento.
12. () Los buses de conexión por su función se clasifican en ATA y SATA.
13. () Los BIOS se encuentran en la memoria ROM.
14. () Los dispositivos mixtos se comportan como dispositivos de entrada y/o salida de datos.
15. () Los buses de datos, transporta los datos en forma unidireccional.
16. () los buses de direcciones, transporta la ubicación a donde llevar los datos en forma unidireccional.
17. () Los buses de control, transporta los comandos y señales del estado de los recursos computacionales.
18. () El mouse, transmite los movimientos de nuestra mano con el computador.
19. () Los “núcleos” de un CPU, son los procesadores que están empaquetados en un único chip, y que realizan su tarea por separado.
20. () La tecnología de CPU multinúcleo ha sido posible gracias al campo de la miniaturización.

Recapitulación del arte

El computador tiene 4 recursos físicos en su estructura, llamados también, elementos del computador, y para su funcionamiento, uno de ellos (los buses de conexión) permite que se comuniquen a través del sistema operativo. El crecimiento exponencial de estos recursos computacionales (procesador, almacenamiento, dispositivos de entrada y salida, y los buses de conexión) nos reafirma que, desde su invención, la computadora ha sido considerada un agente de racionalización y perfeccionamiento; y la creatividad artística representa la lógica y la eficiencia del proceso evolutivo de las personas.

CAPÍTULO II

ANÁLISIS DEL ENUNCIADO

Entrenar y manejar tu mente es la habilidad más importante que debes tener para alcanzar la felicidad y el éxito.

T. Harv Eker.

Para aprender a usar los recursos computacionales requiere que el programador sea creativo. Esta creatividad humana llamada también algoritmo implica saber describir y comunicar los enunciados que reflejan lo que el computador debe dar como resultado del proceso de los datos.

Los enunciados planteados deben ser claros y precisos en función de lo que el problema requiera.

Como observamos en la figura 7, la estructura del arte para analizar los enunciados planteados, partimos desde el enunciado del problema (claro y preciso), al igual que el algoritmo (claro, preciso y finito), y el programa (en lenguaje entendible por la computadora) resolverá el problema tal cual genere los resultados esperados.

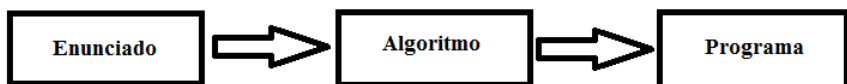


Figura 7. Estructura del arte para analizar los enunciados planteados

2.1. Enunciado de un problema

El enunciado es un conjunto de palabras según el contexto.

Ejemplo: “Determinar la suma de los N primeros números enteros aplicando la siguiente fórmula:

$$\text{Suma} = (N * (N + 1))/2”$$

El enunciado de un problema es un conjunto de palabras que expresan el problema planteado, claros y precisos.

Ejemplo: “En una institución se desea controlar el ingreso de donaciones de acuerdo a su procedencia. Debe existir un registro de donaciones nacionales y otro registro de donaciones del extranjero. Imprimir la lista de cada registro de donaciones, donde al final del reporte aparezca el total donado valorizado en soles, respectivamente”.

Ejemplo: “Elaborar un programa que permita al cantante Juan Gabriel heredar su fortuna, quién antes de morir contabiliza el total de su fortuna, valorizadas en: propiedades aproximadamente 12 millones de dólares; patentes de sus canciones, 45 millones de dólares; joyas y otros aproximadamente 3 millones de dólares depositadas en los bancos. Y decide heredar su fortuna de la siguiente manera: El 30% para cada uno de sus 2 hijos; el 10% para cada uno de sus 2 nietos; el sobrante donará a instituciones sociales,

de las cuales el 50% a la casa de la cultura, el resto donará en partes iguales a los cuatro centros de asistencia social de la comunidad.

Mostrar por pantalla cuánto le corresponde a:

María y Juan (hijos)

Sofía y Pedro (nietos)

Casa de la Cultura

Beneficencia pública (centro de asistencia social)

Centro Adulto Mayor (centro de asistencia social)

Centro de Rehabilitación de Drogadictos (centro de asistencia social)

Centro de Rehabilitación de Discapacitados (centro de asistencia social)”

Ejemplo: “Imprimir los productos que existen en el almacén, en forma ascendente por nombre del producto. Por cada producto debe imprimirse, el nombre del producto, la fecha de ingreso a almacén, la cantidad existente en almacén, su precio unitario del producto a la fecha de adquisición, su unidad de medida y precio unitario de venta del producto”.

Ejemplo: “El usuario sólo de lunes a viernes puede imprimir en forma ascendente por apellido paterno, la lista de beneficiarios de la deuda social del sector educación del Gobierno Regional de San Martín, considerando: número correlativo, nombres y apellidos del beneficiario y el monto de la deuda, respectivamente”.

Ejemplo: “El programa debe automatizar la venta de licores al usuario autorizado, donde el cliente por más de S/. 250 de compra puede elegir de regalo uno de los tres licores peruanos ofertados (vino el abuelo, pisco o ron). En el ticket de compra, además de los licores comprado, imprimir al final de página, el nombre del licor peruano entregado”.

Ejemplo: “Mostrar en pantalla de forma ascendente, nombres y apellidos de los 35 alumnos matriculados en la asignatura de Filosofía”.

“Ejemplo: “Un restaurant ofrece un descuento del 10% para consumos de hasta S/. 30.00; un descuento de 20% para consumos mayores. Para ambos casos se aplica el impuesto general a las ventas de 18%. Imprimir el comprobante de pago por lo consumido, incluyendo datos generales del cliente (DNI, nombre del cliente, dirección y fecha), el detalle por consumo (número correlativo, descripción, cantidad, unidad de medida e importe) y los totales (total descuento, total impuestos y total general)”.

Ejemplo: “Automatizar el descuento que le corresponde sobre el total del monto comprado. Compras para un monto hasta 50 soles se aplica un descuento de 10%. Y compras por un monto mayor a 50 soles se aplica un descuento de 20%. Imprimir el importe a pagar”.

Ejemplo: “Ingresar y mostrar en pantalla el título y el autor del libro buscado en la Biblioteca de la Municipalidad Provincial de San Martín”.

2.2. Análisis del enunciado

Analizar el enunciado planteado de un problema cualquiera que sea su origen, requiere entender con claridad y precisión el problema. Este paso es fundamental para cualquier programador, porque rara vez encontraremos iguales soluciones a un problema en particular. Inclusive, un buen análisis del enunciado puede orientar al título del algoritmo y por ende al título del programa.

Por eso es muy importante que el enunciado del problema sea validado por todas las personas involucradas. Ojo, no inventar, ni suponer cosas que no lo diga el enunciado.

El programador, cuanto más hace ese ejercicio de formular enunciados de problemas acumulara experiencia, destrezas y por ende menos tiempo.

El propósito del enunciado es llegar a comprender la naturaleza del problema, por lo que, el enunciado requiere que las especificaciones de entrada y salida sean descritas con detalle, precisión y llegar a un resultado satisfactorio.

Según Joyanes L. (1988). El análisis del problema exige una lectura previa del problema a fin de obtener una idea general de lo que se solicita. La segunda lectura deberá servir para responder a las preguntas:

¿Qué información debe proporcionar la resolución del problema?

¿Qué datos se necesita para resolver el problema?

La respuesta a la primera pregunta indicará los resultados deseados o las salidas del problema. La respuesta a la segunda pregunta indicará qué datos se proporcionan o las entradas del problema.

De lo anterior, en el siguiente capítulo “Datos identificados para el arte” se aplica el análisis del enunciado en detalle.

Preguntas de repaso

Marcar verdadero (V) o falso (F)

1. () El enunciado debe ser validado por todas las personas involucradas.
2. () El enunciado es un conjunto de palabras que expresan el problema planteado.
3. () Los enunciados planteados deben ser claros y precisos.
4. () Un buen análisis del enunciado puede orientar al título del programa.
5. () Aprender a usar los recursos computacionales requiere que el programador sea creativo.
6. () El enunciado refleja lo que el computador debe dar como resultado del proceso de los datos.
7. () La creatividad humana debe ser bien descrita.
8. () El enunciado refleja lo que el computador debe procesar.
9. () La creatividad humana debe comunicarse a través de los enunciados.
10. () Un buen análisis del enunciado permite una efectiva valoración del arte para programar computadoras.

Plantear cinco enunciados de problemas relevantes en el entorno donde se encuentre.

11. Relacionado al control de almacenes.
12. Relacionado al proceso de ventas en una farmacia.
13. Relacionado a la planificación de actividades de proyectos.
14. Relacionado al seguimiento de un proyecto de inversión pública.
15. Relacionado al proceso de compras de materiales para ejecución de una obra.

Recapitulación del arte

El enunciado del problema que queremos resolver usando el computador debe ser claro y preciso, validado por todas las personas involucradas en la solución. Esto con el propósito de resolver el problema y generar los resultados esperados por todos. Inclusive, un buen análisis del enunciado, puede orientar al título del algoritmo y por ende al título del programa.

CAPÍTULO III: DATOS IDENTIFICADOS PARA EL ARTE

Los algoritmos son capaces de emular la creatividad.

Hugo Caselles-Dupré De Obvious

Los datos identificados en los enunciados permiten hacer el análisis previo al proceso de transformación. Las cuales darán los resultados esperados.

En la figura 8, observamos en la estructura del arte para programar computadoras, tanto los datos de entrada como los datos de salida al ser identificados en el enunciado deben respetarse rigurosamente y no inventarlos.



Figura 8. Estructura del arte para programar computadoras

3.1. Datos para el arte

Los datos identificados en el enunciado se clasifican en datos de entrada y datos de salida.

Todos los datos identificados en el enunciado del problema permiten planificar la solución (elaborar el algoritmo), estos se agrupan en:

3.1.1. Datos de entrada

Los datos de entrada son datos para ser transformados.

3.1.2. Datos de salida

Los datos de salida son los datos que representan a los resultados esperados.

3.1.3. Datos mixtos

Los datos mixtos llamados así a los datos que como resultado del análisis del enunciado pueden estar datos de entrada y datos de salida.

3.2. Identificadores

Los identificadores son los nombres que se les asigna a los objetos, los cuales pueden ser variables o constantes.

Los identificadores o nombres significativos de las variables se sugiere que representen el contenido o valores que albergará y no coincidir con palabras reservadas.

3.3. Palabras reservadas

Las palabras reservadas son palabras que representan a un identificador de acción, como: inicio, fin, mientras, hacer, etc.

También existen palabras reservadas relacionadas a las funciones internas que realizan operaciones matemáticas, como:

$\text{abs}(x)$: valor absoluto de x .

$\text{arctan}(x)$: arco tangente de x .

$\text{cos}(x)$: coseno de x .

$\text{exp}(x)$: exponencial de x .

$\text{ln}(x)$: logaritmo neperiano de x .

$\text{log10}(x)$: logaritmo decimal de x .

$\text{redondeo}(x)$: redondeo de x .

$\text{seno}(x)$: seno de x .

$\text{cuadrado}(x)$: cuadrado de x .

$\text{raizcua}(x)$: raíz cuadrada de x .

$\text{trunc}(x)$: truncamiento de x .

3.4. Tipos de datos

En cuanto a los tipos de datos identificados en el enunciado del problema, tenemos:

3.4.1. Datos numéricos

Los datos numéricos representan un valor entero o real.

Enteros, es un subconjunto finito de los números enteros y pueden ser positivo o negativo.

Ejemplo: 3, -7, 40, -11

Reales, es un subconjunto de números reales y tienen un punto decimal. Pueden ser positivo o negativo.

Ejemplo: 0.003, -7.51, 40.89, -111.99

3.4.2. Datos lógicos

Los datos lógicos o booleano solo puede tomar uno de los dos valores: verdadero o falso. También son utilizados para representar las alternativas (si/no) condicionales.

3.4.3. Dato tipo carácter

Dato tipo carácter tiene como valor un carácter. Pueden ser:

Carácter alfabético: A, a, B, b, ..., Z, z.

Carácter numérico: 1, 2, 3, ..., 9.

Carácter especial: +, -, <, >, ..., *, /, ...

3.4.4. Datos tipo cadena de caracteres

Los datos tipo cadena de caracteres es una sucesión de caracteres delimitados por una comilla (apóstrofo) o doble comillas.

Caracteres alfabéticos: “Hola Aarón”

Caracteres numéricos: “28 2019”

Caracteres alfanuméricos: “28 de febrero de 2019”

Caracteres especiales: “+, -, *, /”

3.5. Variables

Las variables son datos cuyo valor puede cambiar durante la ejecución del algoritmo.

Una variable tiene un identificador que está asociado a un espacio en memoria y almacena un valor.

Por su uso las variables pueden ser:

Variable local, llamado así cuando la variable sólo es utilizada en parte del algoritmo.

Variable global, llamado así cuando la variable es utilizada en todo el algoritmo.

3.6. Constantes

Las constantes son datos cuyo valor no cambia durante la ejecución del algoritmo.

Una constante tiene un identificador que está asociado a un espacio en memoria y almacena un valor.

3.7. Contadores

Los contadores son variables cuyo valor se asigna en función del incremento o decremento de las repeticiones durante la ejecución del algoritmo.

3.8. Acumuladores

Los acumuladores son variables cuyo valor puede aumentar o disminuir durante la ejecución del algoritmo.

3.9. Operadores

Los operadores están directamente relacionados a las expresiones que usan las combinaciones de constantes, variables, símbolos de operación, paréntesis y nombres de funciones o procedimientos durante la ejecución del algoritmo.

3.9.1. Operadores aritméticos

Los operadores aritméticos tienen similitud a los operadores matemáticos y se usan en las expresiones aritméticas con valores de variables y constantes numéricas, respectivamente.

+ : suma.

- : resta.

* : multiplicación.

/ : división.

% : módulo, el resto después de la división.

** : potencia o exponenciación.

++ : incremento.

-- : decremento.

div : división entera.

3.9.2. Operadores relacionales o de comparación

Los operadores relaciones se usan para las operaciones binarias de comparación.

== : igual que.

!= : diferente que.

< : menor que.

> : mayor que.

<= : menor o igual que

>= : mayor o igual que

3.9.3. Operadores lógicos o booleanos

Los operadores lógicos o booleanos, llamado así en honor del matemático británico George Boole, quien desarrollo el álgebra lógica o de *Boole*, son operadores de comparación y usan las siguientes palabras reservadas:

OR : " O " Devuelve VERDADERO si uno de los operadores es VERDADERO.

AND : " Y " Devuelve VERDADERO si ambos operadores son VERDADERO.

NOT : "No" Devuelve VERDADERO si la negación del operando es VERDADERO.

3.9.4. Operadores de asignación

Los operadores de asignación son la forma de darles valores a una variable. Se representa con los signos o símbolos:

= : Igual

En los operadores de asignación no se debe asignar valores diferentes al tipo de datos que corresponde a las variables.

Los operadores de asignación están estrechamente ligadas a la instrucción de declaración de variables.

3.10. Expresión

Según Joyanes L., Rodríguez L., & Fernández M. (1996), una expresión es una combinación de operadores y operandos. Los operandos podrán ser constantes, variables u otras expresiones y los operadores de cadena, aritméticos, relacionales o lógicos. Las expresiones se clasifican, según el resultado que producen, en: numéricas, alfanuméricas o booleanas.

3.11. Ejemplos

Ejemplo: Dado el siguiente enunciado, identificar los datos de entrada y los datos de salida.

“Para usar el sistema informático, la persona debe ingresar un usuario y clave (asignar un valor a cada uno); si el usuario y la clave es correcto, mostrar por pantalla: “Bienvenido al Sistema”, nombres y apellidos; y si no es correcto, mostrar por pantalla: “Gracias por intentarlo” ”.

Identificación de los datos de entrada:

Nombres

Apellidos

Usuario

Clave

Mensaje de Usuario y Clave Correcto

Mensaje de Usuario y Clave No Correcto

Login

Password

Identificación de los datos de salida:

Mensaje de Usuario y Clave Correcto

Mensaje de Usuario y Clave No Correcto

Nombres

Apellidos

Ejemplo: Dado el siguiente enunciado, identificar los datos de entrada y los datos de salida, declarar variables, declarar constantes, identificar los operadores aritméticos, operadores de comparación y operadores lógicos.

“La persona usuaria del sistema informático, debe ingresar un usuario y clave (asignar un valor a cada uno); si el usuario y la clave no es correcto, mostrar por pantalla: “Gracias por intentarlo”. Y si es correcto, mostrar por pantalla el monto a pagar por la venta de un producto que cuesta S/. 16.00 el kilo o el litro, considerando lo siguiente: (1) Descuento del 10% por la venta del producto hasta 30 kilos o litros. (2) Descuento del 15% por la venta del producto por más de 30 kilos o litros. (3) En el caso de no existir el producto para la venta en kilos completar la venta del producto en litros, previa conformidad por el cliente”.

Identificación de los datos de entrada:

Login

Password

Usuario

Clave

Mensaje de Usuario y Clave No Correcto

Cantidad del producto existente en litros

Cantidad del producto existente en kilos

Nombre del producto

Monto a pagar

Descuento del 10 %

Descuento del 15 %

Cantidad a comprar

Cantidad faltante de lo existente

Conformidad del cliente

Identificación de los datos de salida:

Monto a pagar

Mensaje de Usuario y Clave No Correcto

Nombre del producto

Declaración de variables:

Usuario

Clave

Monto_pagar

Producto

Falta

Acepta

Declaración de constantes:

Login

Password

Mensaje

Cantidad_Producto_Litros

Cantidad_Producto_Kilos

Descuento_10

Descuento_15

Identificación de operadores aritméticos:

- + suma
- * multiplicación
- resta

Identificación de operadores de comparación:

- == igual que
- <= menor o igual que

Identificación de operadores lógicos:

AND

Ejemplo: Dado el siguiente enunciado, identificar los datos de entrada y los datos de salida, declarar variables, declarar constantes, identificar los operadores aritméticos, operadores de comparación y operadores lógicos.

“La persona usuaria del sistema informático, debe ingresar un usuario y clave (asignar un valor a cada uno); si el usuario y la clave no es correcto, mostrar por pantalla: “Usuario o clave incorrecto”. Y si es correcto, mostrar por pantalla los nombres y apellidos del usuario. El sistema informático permite automatizar la venta de productos ofertados en la promoción por aniversario de la Bodega “Cinco Esquinas”. La promoción consiste en entregar vales de compra al cliente según resultado obtenido en el juego de la ruleta, cuyos valores equivalen a los del reloj y se realiza en forma manual. Hay vales de compra desde cero hasta 12 puntos, consignados de la siguiente manera: (1) Vale de compra por S/.

10.00 de cero hasta un punto. (2) Vales de compra por S/. 20.00 entre uno hasta dos puntos. (3) Vales de compra por S/. 30.00 de dos hasta tres puntos. (4) Vales de compra por S/. 40.00 de tres hasta cuatro puntos. (5) Vales de compra por S/. 50.00 de cuatro hasta cinco puntos. (6) Vales de compra por S/. 60.00 de cinco hasta seis puntos. (7) Vales de compra por S/. 70.00 de seis hasta siete puntos. (8) Vales de compra por S/. 80.00 de siete hasta ocho puntos. (9) Vales de compra por S/. 90.00 de ocho hasta nueve puntos. (10) Vales de compra por S/. 100.00 de nueve hasta diez puntos. (11) Vales de compra por S/. 110.00 de diez hasta once puntos. (12) Vales de compra por S/. 120.00 de once hasta doce puntos.

Para aplicar el vale de compra, el monto total a pagar debe ser cinco veces mayor o igual al valor del vale de compra. El vale de compra es el descuento del monto total a pagar.

Al final, mostrar el monto total a pagar”.

Identificación de los datos de entrada:

Usuario

Clave

Login

Password

nombres

apellidos

vale de compra
monto a comprar
monto a pagar
conformidad del cliente

Identificación de los datos de salida:

nombres
apellidos
monto a pagar
mensaje de error

Declaración de variables:

usuario
clave
vale
monto
monto_pagar
conformidad

Declaración de constantes:

login
password
nombres
apellidos

Identificación de operadores aritméticos:

- resta

* multiplicación

Identificación de operadores de comparación:

== igual que

>= mayor o igual que

Identificación de operadores lógicos:

AND

Preguntas de repaso

Marcar verdadero (V) o falso (F)

1. () Los datos identificados en el enunciados deben ser las únicas no inventarlos.
2. () Los operadores lógicos son operadores de comparación.
3. () Los valores en los operadores de asignación son diferentes al tipo de datos que corresponde a las variables.
4. () Los operadores están directamente relacionados a las expresiones que usan las combinaciones de constantes, variables, símbolos de operación, paréntesis y nombres de funciones o procedimientos durante la ejecución del algoritmo.
5. () Las palabras reservadas representan a un identificador de acción(inicio, fin, mientras, etc.)
6. () Los contadores son variables cuyo valor se asigna en función del incremento o decremento de las repeticiones durante la ejecución del algoritmo.

7. () Las constantes son datos cuyo valor no cambia durante la ejecución del algoritmo.
8. () Las variables son datos cuyo valor puede cambiar durante la ejecución del algoritmo.
9. () Los datos de entrada son los datos que representan a los resultados esperados.

Por cada enunciado:

- Identificar Datos de Entrada y Salida.
 - Declarar variables y/o constantes.
 - Identificar operadores aritméticos, operadores de comparación y operadores lógicos.
 - Elaborar el Diagrama de Flujo de Datos.
10. “El cliente elige uno de los tres platos ofertados en el restaurante (arroz con pollo, lomo saltado y tallarines verdes), y por cada S/. 10 de consumo tiene un descuento de 5% del costo del plato. Mostrar el monto a pagar por cliente”.
 11. “Luis es un antiguo granjero, hace poco hizo empollar 13,640 huevos y nacieron unos hermosos pollitos, él quiere ponerlas en 12 galpones. ¿Cuántos pollitos pondrá en cada galpón?. ¿Cuántos pollitos sobran?”.
 12. “El cliente compra un celular y el vendedor debe entregar un reporte conteniendo la marca del celular, el nombre del cliente,

el pago mensual y el plan de contrato a la pertenece, sabiendo que la empresa operadora de comunicaciones oferta tres tipos de planes: Plan A de S/. 59, Plan B de S/. 69 y Plan C de S/. 79”.

13. “Todas las personas colaboran en una coleta. ¿Cuánto de dinero se recolecta?”.
14. “Del total de personas. ¿Cuántos son varones y cuántos son mujeres?”.
15. “Mostrar por pantalla todas las personas que colaboraron para los damnificados de la Región Piura”.
16. “Imprimir la lista de damnificados de la Ciudad de Catacaos. En la cabecera del reporte debe incluir el título, la fecha y la hora de impresión”.
17. “El reporte debe contener, en la cabecera el título del reporte, fecha y hora de impresión. En el pie de página, el nombre de la empresa y el e-mail del Gerente”.
18. “Mostrar por pantalla los nombres y apellidos de los Ingenieros adscritos al Departamento Académico de Sistemas e Informática”.
19. “El reporte a imprimir debe contener el número de orden correlativo, nombre del producto y el precio unitario del producto”.

20. “La lista a mostrar por pantalla debe contener nombres y apellidos, edad y sexo, de los alumnos de la Facultad de Ingeniería de Sistemas e Informática, respectivamente”.
21. “Imprimir los nombres, apellidos y número de camiseta, de los jugadores de cada equipo de fútbol del partido a disputar en el Estadio Nacional del Perú”.
22. “Para cada Director del Gobierno Regional de San Martín, mostrar por pantalla, sus nombres y apellidos, edad, estado civil, profesión y tiempo de servicio”.

Rescapitulación del arte

Los datos identificados en los enunciados son los únicos a tener en cuenta para el proceso de transformación que realiza el computador y generar los resultados esperados. Estos datos (numérico, texto o booleano) deben respetarse rigurosamente y no inventarlos. Los operadores determinan el tipo de operación a realizar por el computador, dependiendo de los resultados a obtener. En la programación estos datos o valores se asignan (con operadores de asignación) a las variables o constantes, que ocupan un espacio de almacenamiento en el computador.

CAPÍTULO IV TÉCNICA PARA EL ARTE

Las computadoras son inútiles. Sólo pueden dar respuestas.

Pablo Picasso, pintor.

Según Rosenblatt D. (2018), los algoritmos pertenecen hoy más que nunca en nuestra vida cotidiana. Nos ayudan a escoger una película, encontrar pareja, responden preguntas, nos señalan el camino a través de la ciudad y a menudo son más precisos que los humanos.

La técnica a usar para el arte en este libro es la técnica de diagrama de flujo de datos.

4.1. Técnica de diagrama de flujo de datos

Los DFD son técnicas o herramientas donde se puede apreciar el valor de colaboración y comunicación efectiva entre las personas que con mucho arte elaboran los algoritmos. Hay que tener creatividad para construir un algoritmo que exprese lo esencial la comprensión del proceso/procedimiento como un dibujo. Obviamente el cerebro humano reconoce más fácilmente los dibujos que la escritura. Un artístico y buen diagrama de flujo de datos reemplaza muchas líneas y páginas de texto. Además, los interesados en conocer el algoritmo (nuevos empleados, etc.)

pueden usar el DFD como una excelente herramienta para capacitación.

Los DFD se elaboran sin importar el lenguaje de programación, que es el siguiente paso para usar los recursos computacionales y resolver el problema que describe el enunciado planteado.

4.2. Símbolos en el DFD

El algoritmo está compuesto de símbolos que se convierten en un estándar de comunicación. Cada símbolo tiene su propio significado lo que garantiza que el algoritmo tenga su interpretación y análisis en forma clara y precisa.



Los símbolos en el DFD son estándares internacionales que para este propósito se han creado.





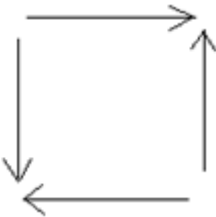


Según Calderón S. & Ortega J. (2009), entre las instituciones internacionales creadores de símbolos a usar para diagramar flujos de datos tenemos:

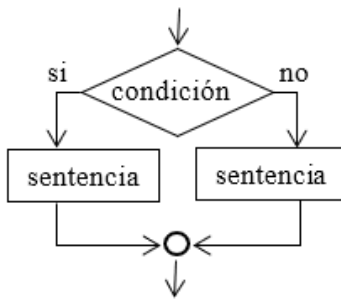
- 1) Simbología ASME (American Society of Mechanical Engineers – Sociedad Americana de Ingenieros Mecánicos).
- 2) Simbología ANSI (American National Standard Institute – Instituto Nacional de Normalización Estadounidense).

- 3) Simbología ISO (International Organization for Standardization – Organización Internacional para la Normalización).
- 4) Simbología DIN (Deutches Institut fur Normung e.v. – Instituto Alemán de Normalización).
- 5) Simbología DO (Simbología del Flujograma de Ingeniería de Operaciones y de Administración y Mejora de la Calidad del Proceso).
- 6) Simbología DIF (Simbología para Diagramas Integrados de Flujo en las versiones de Yourdon De Marco y Gene & Sarson).

Los símbolos considerados en este libro se muestran en la figura 9, son los que usaremos para elaborar los algoritmos.

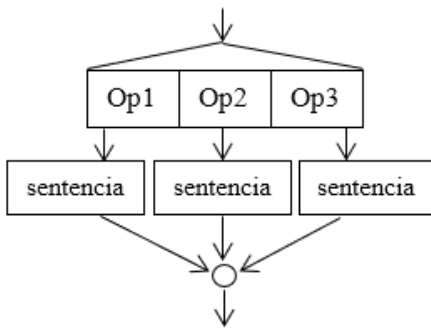
Símbolo	Nombre	Descripción
	Inicio/Fin	Indica el inicio o fin del diagrama.
	Entrada/Salida de datos	Permite la entrada o salida de datos y que sea procesada por un periférico.

	Proceso	Permite procesos, ya sea una instrucción de asignación u operación aritmética.
	Salida de datos por impresora	Indica la salida de datos por impresora.
	Lectura de datos desde teclado	Representa entrada de datos registradas por teclado.
	Salida de datos por pantalla	Indica la salida de datos por pantalla o consola.
	Flujo de datos	Dirección de flujo de datos. Indica la secuencia que se realizan las operaciones.
	Conector de diagrama	Representa la continuidad del diagrama dentro de la misma página.
	Conector de página	Representa la continuidad del diagrama en otra página.



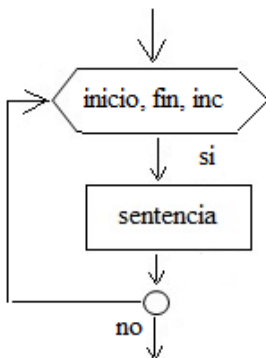
Decisión simple

Permite la toma de decisión en base a una condición lógica o de comparación entre datos



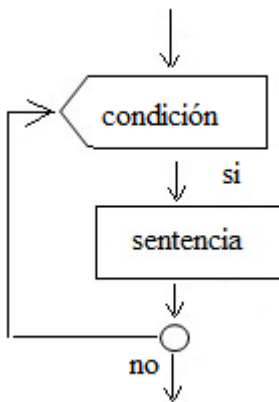
Selección o decisión múltiple

Dada una expresión que se compara, permite escoger una opción de muchas.



Estructura para

Símbolo para esta estructura de control repetitiva se usa generalmente cuando se conoce de antemano el



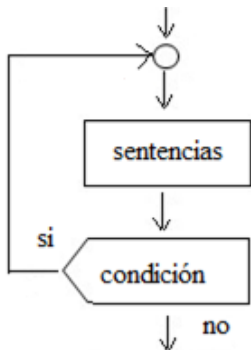
Estructura
mientras

número de
iteraciones.

Posee 3

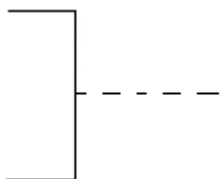
parámetros:
inicio, fin e
incremento.

Símbolo hasta
mientras. Dada
una expresión
al principio de
la iteración
esta es
evaluada; si la
condición es
verdadera
realizará el
ciclo, si es
falsa la
repetición
cesará. La
iteración es
manual.



Estructura
hacer
mientras

Funciona igual que la estructura Mientras, con la diferencia que al menos una vez hará el grupo de instrucciones y luego evaluará una condición. Si la condición evaluada es verdadera continúa dentro del ciclo y si es falsa termina la iteración. La iteración es manual.



Comentarios

Se utiliza para añadir comentarios clasificadores a otros símbolos del diagrama de flujo.

Figura 9. Símbolos usados en el DFD

Los símbolos pueden estandarizarse institucionalmente.

4.3. Construcción de los DFD

La construcción de los DFD requiere una serie de pasos que sirven de guía para su diseño, y mucha disciplina reflejado en el uso de la simbología, individual o colectivamente (política de la institución). La secuencia de pasos sugerido, son:







- Definir el título del DFD. El título debe representar al contenido del enunciado planteado.
- Identificar los datos de entrada, datos de salida y datos mixtos. Estos datos están en el enunciado, no suponer.
- Definir identificadores o nombres de las variables y constantes. Pueden ser nombres cortos o nombres largos.
- Identificar los operadores u operaciones a usar en el DFD.
- El diseño del algoritmo consta de INICIO, declaración de variables y constantes, el procedimiento (sentencias) que es el cuerpo del diagrama, y FIN.
- Describir los pasos del procedimiento especificando con frases cortas pero completas. Las frases deben comenzar con un verbo en tercera persona del singular del tiempo presente indicativo. Ejemplo: Asigna, controla, remite, recibe, etc. No es conveniente que las frases lleven términos técnicos.

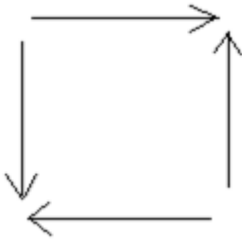
Según Pinales F. & Velázquez C. (2014), Los diagramas de flujo son una herramienta que permite representar visualmente qué operaciones se requieren y en qué secuencia se deben efectuar para solucionar un problema dado. Por consiguiente, un diagrama de flujo es la representación gráfica mediante símbolos especiales, de

los pasos o procedimientos de manera secuencial y lógica que se deben realizar para solucionar un problema dado.

Preguntas de repaso

Hacer corresponder con una flecha el nombre con el símbolo del diagrama de flujo de datos.

Símbolo	Nombre
	Flujo de datos
	Proceso
	Entrada/Salida de datos
	Selección o decisión múltiple
	Estructura para
	Decisión simple



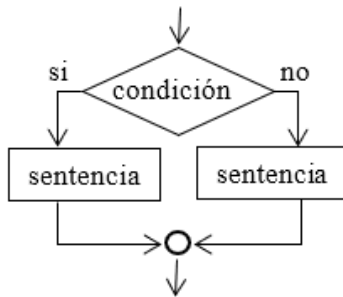
Inicio/Fin



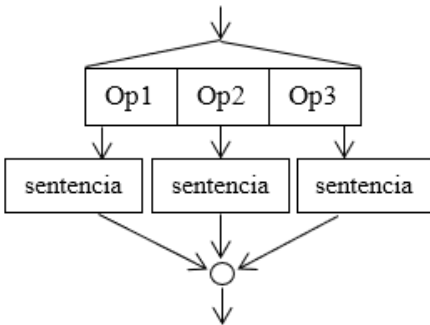
Conector de página



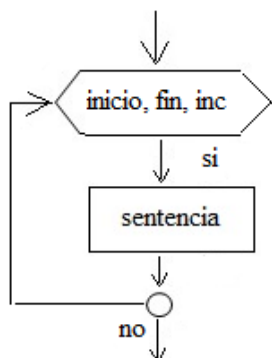
Comentarios



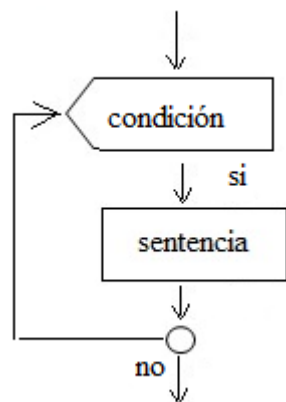
Salida de datos por pantalla



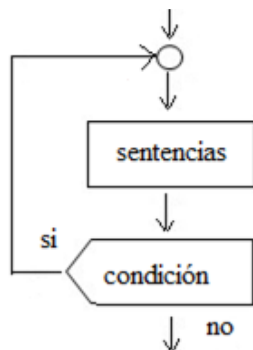
Salida de datos por impresora



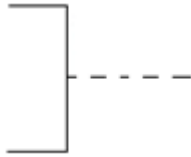
Lectura de datos desde teclado



Estructura hacer mientras



Estructura mientras



Conector de diagrama

Recapitulación del arte

El DFD describe gráficamente un proceso. Permite representar visualmente qué operaciones se requieren y en qué secuencia se debe efectuar para implementar la solución a un enunciado. Es una forma de hacer arte. La creatividad en su máxima expresión usando símbolos o notaciones que describen el contenido de un enunciado planteado. El DFD describe y comunica el planteamiento del enunciado sin importar el lenguaje de programación a usar para ser compilado o interpretado por el computador. La construcción de los DFD requiere una serie de pasos que sirven de guía para su diseño, y mucha disciplina reflejado en el uso de la simbología, individual o colectivamente (política de la institución).

CAPÍTULO V DISEÑO DEL ARTE

Todo el mundo en este país debería aprender a programar un computador ... porque te enseña a pensar.

Steve Jobs.

Según Revista sobredosis (2018). “Somos artistas con un tipo diferente de pincel. Nuestro pincel es un algoritmo desarrollado en una computadora”, dijo Hugo Caselles-Dupre, un ingeniero informático que fundó el grupo con amigos de la infancia, Gauthier Vernier y Pierre Fautrel.

5.1. Estructura secuencial

Estructura secuencial, después de la declaración de variables, consiste en ejecutar una tras otra las sentencias, sin ninguna condición, como observamos en la figura 10.

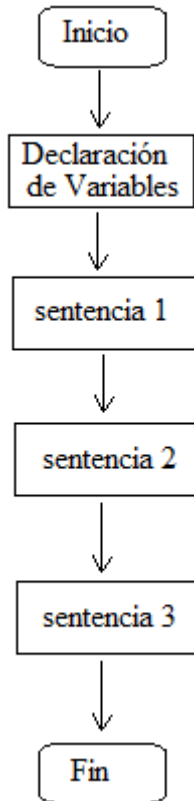


Figura 10. DFD de estructura secuencial.

Ejemplo: Elaborar el DFD del siguiente enunciado: “Al final del día, en el comedor universitario se reporta el nombre del alumno, fecha y hora que recibe el servicio”.

En la figura 11, se muestra el DFD para obtener el reporte de servicios del comedor universitario.

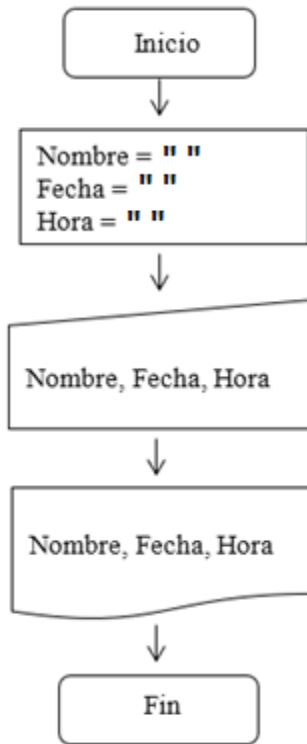


Figura 11. DFD Reporte de servicios del comedor universitario

Ejemplo: Elaborar el DFD del siguiente enunciado: “Ingresar y mostrar en pantalla el título y autor de los libros de la Biblioteca”.

En la figura 12, se muestra el DFD para obtener el reporte de libros de la biblioteca.

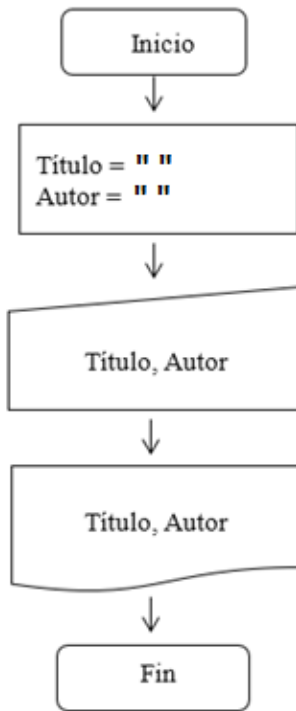


Figura 12. DFD libros de la Biblioteca

5.2. Estructura de control de decisión simple

Estructura de control de decisión simple, después de la declaración de variables, consiste en ejecutar sentencias en la dirección de las flechas y tomar decisiones de por lo menos uno en DFD en base a una condición lógica o de comparación de datos, como observamos en la figura 13.

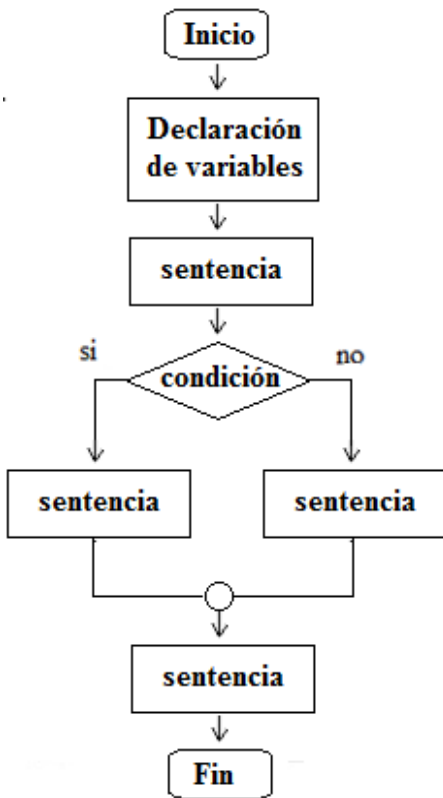


Figura 13. DFD de control de decisión simple

Ejemplo: Elaborar el DFD del siguiente enunciado: “Las mujeres con hijos reciben la donación de víveres de la Oficina de Bienestar Social. Imprimir la ficha de donación”

En la figura 14 se muestra el DFD para generar el reporte de entrega de víveres.

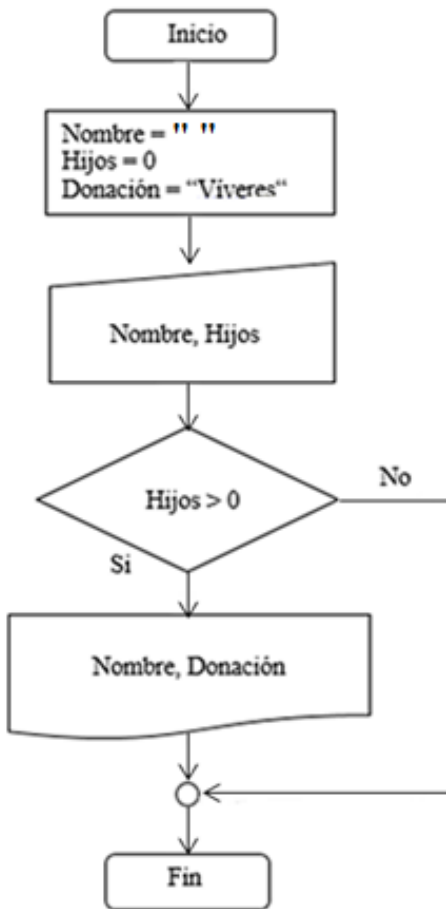


Figura 14. DFD de reporte de entrega de víveres

Ejemplo: En la figura 15, se muestra el DFD del siguiente enunciado: “Los alumnos que se matriculan en 21 o más créditos por ciclo académico son considerados alumnos regulares”.

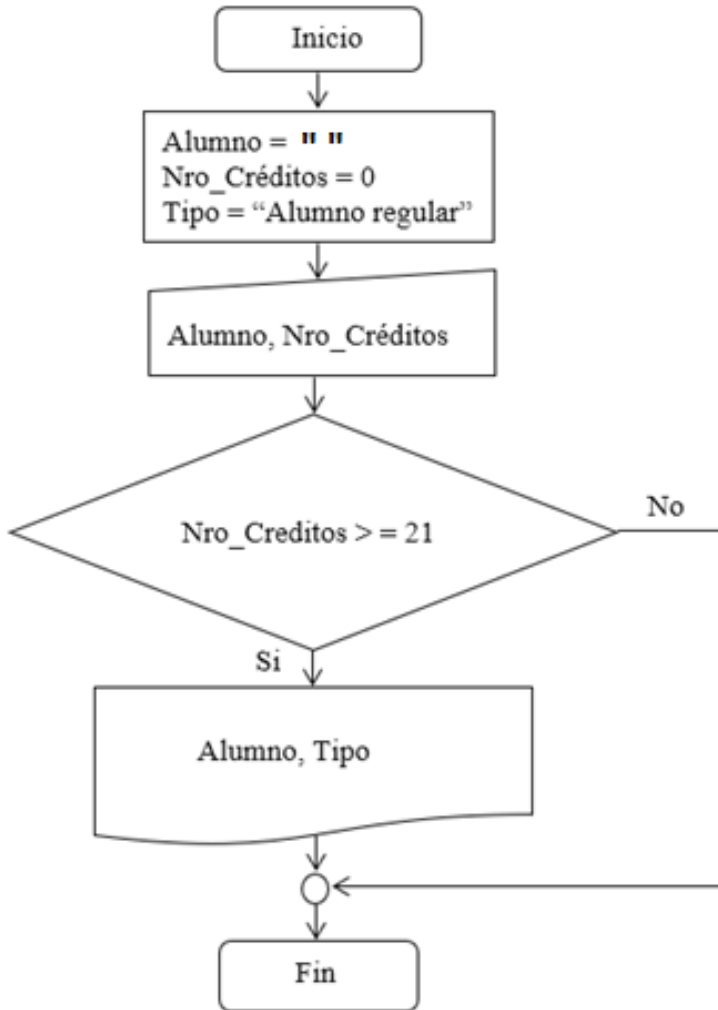


Figura 15. DFD Tipo de matrícula

5.3. Estructura de control de decisión múltiple

Estructura de control de decisión múltiple, después de la declaración de variables, consiste en ejecutar sentencias en la dirección de las flechas y dada una expresión o selector es evaluada una a una hasta encontrar una opción de muchas que cumpla el valor del selector y ejecutar las sentencias, y termina. Tal como se muestra en la figura 16.

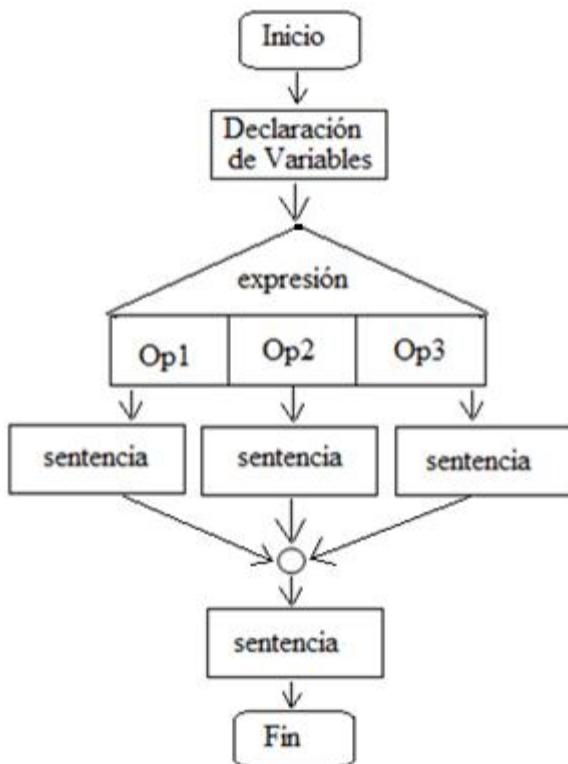


Figura 16. DFD de control de decisión múltiple

Ejemplo: En la figura 17 muestra el DFD del siguiente enunciado: “Mostrar por pantalla las siguientes opciones: 1.- Nuevo; 2.- Abrir; 3.- Grabar 4.- Cerrar. Luego seleccionar la opción a realizar. Si la opción seleccionada es 1 debe mostrar en pantalla el mensaje “Nuevo archivo”. Si la opción seleccionada es 2 debe mostrar en pantalla el mensaje “Abrir archivo”. Si la opción seleccionada es 3 debe mostrar en pantalla el mensaje “Grabar archivo”. Si la opción seleccionada es 4 debe mostrar en pantalla el mensaje “Cerrar archivo””.

Ejemplo: En la figura 18 muestra el DFD del siguiente enunciado: “En una tienda, se debe controlar las cuatro formas de pago: (1) En efectivo; (2) Con tarjeta; (3) Canje; y (4) Con tarjeta y en efectivo; y se calcula multiplicando el monto por la cantidad e imprime el monto a pagar”. La figura 18 muestra el DFD de forma de pago.

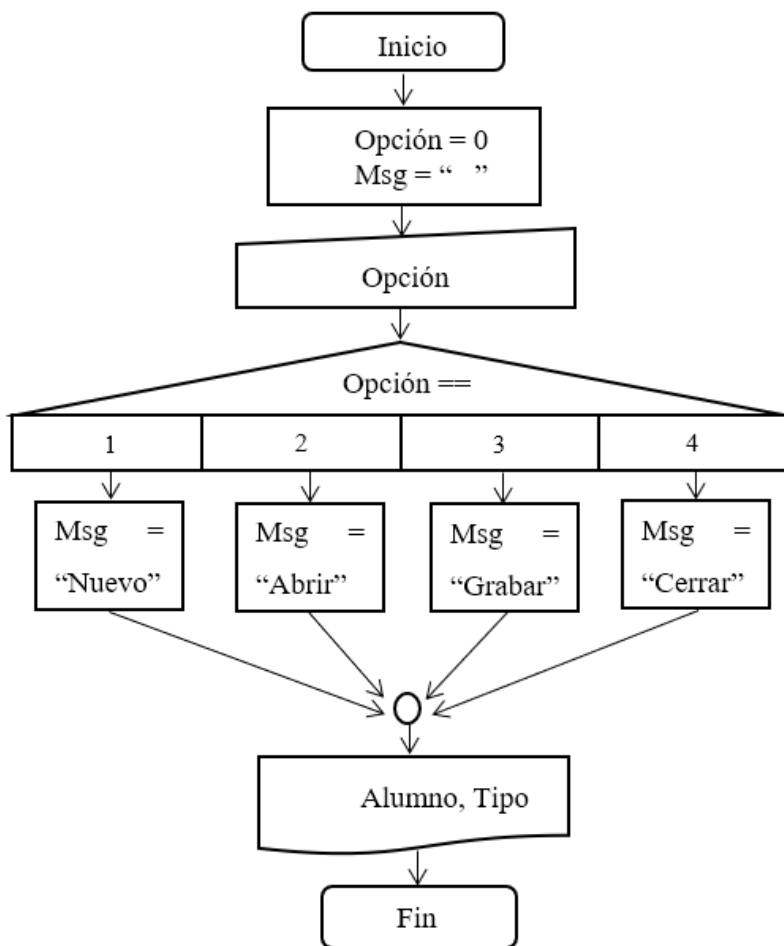


Figura 17. DFD Elige opción

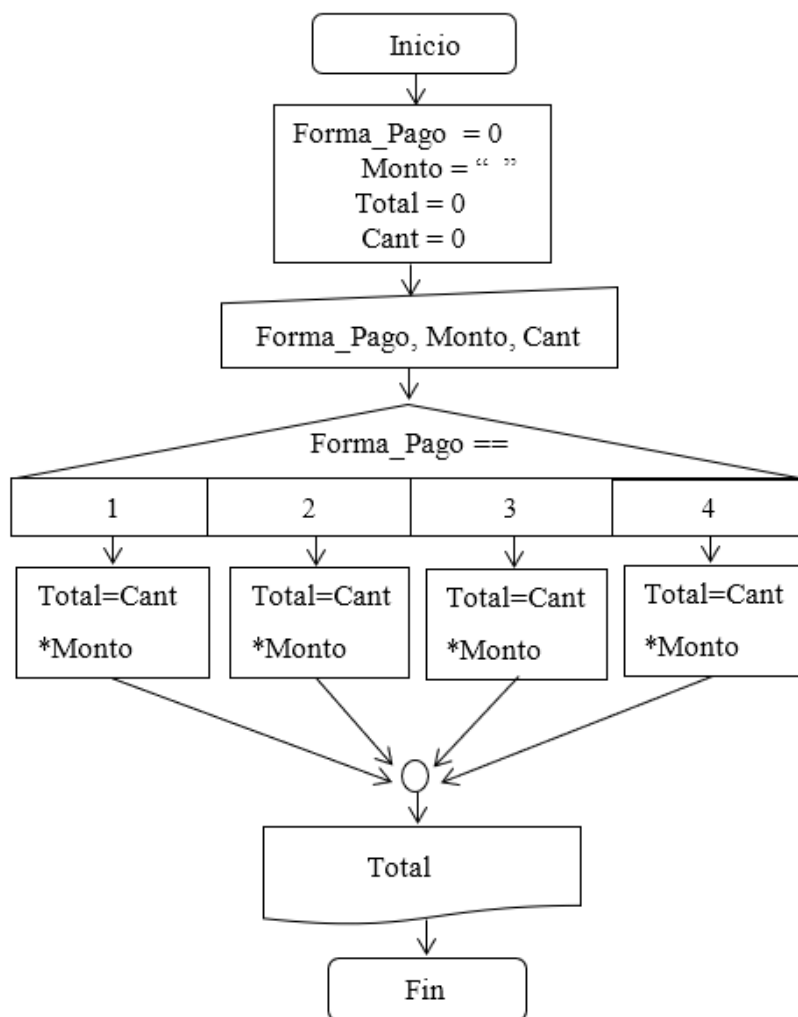


Figura 18. DFD Elige forma de pago

5.4. Repeticiones

5.4.1. Estructura PARA

En la figura 19, después de la declaración de variables, se ejecuta las sentencias en la dirección de las flechas, y se usa la estructura PARA si se conoce de antemano el número de repeticiones en forma automática donde el contador (variable) posee tres parámetros que son valor inicial, valor final e incremento o decremento, y el número de repeticiones de la estructura PARA lo controla el contador y luego termina.

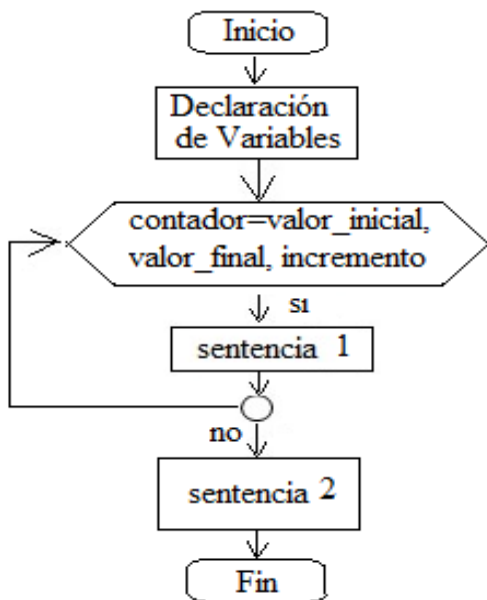


Figura 19. DFD de la estructura PARA

Ejemplo: En la figura 20, se muestra el DFD del siguiente enunciado: “Imprimir el índice de la revista de diez reportajes, conteniendo: título, autor, contenido, fecha de edición del reportaje.”.

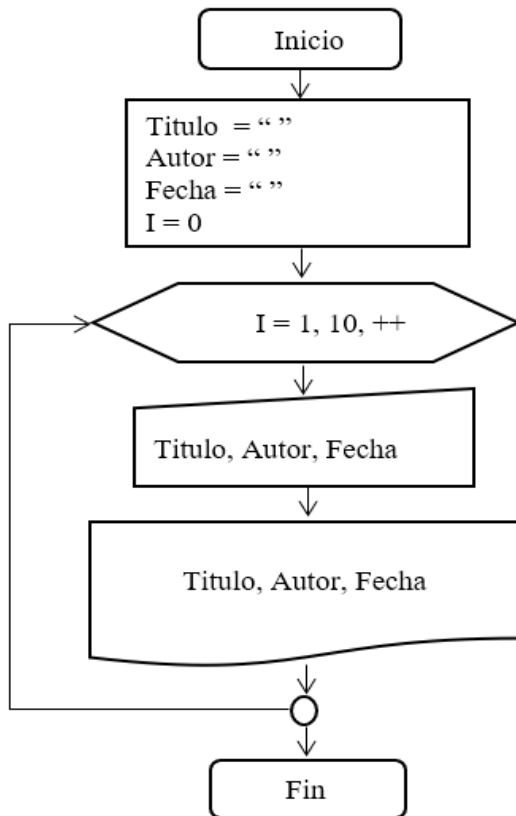


Figura 20. DFD Índice de revista deportiva

Ejemplo: En la figura 21, se muestra el DFD del siguiente enunciado: “Mostrar nombres y apellidos de los 35 alumnos matriculados en la asignatura de Filosofía”.

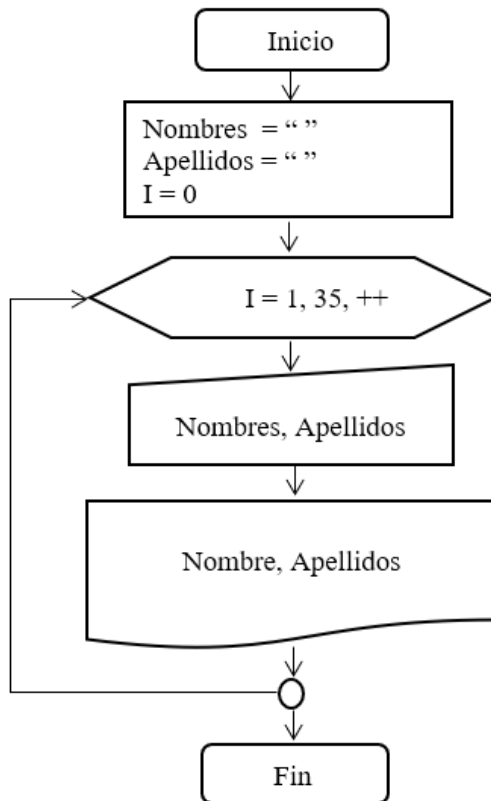


Figura 21. DFD Lista de alumnos de Filosofía

5.4.2. Estructura MIENTRAS

En la figura 22, después de la declaración de variables, consiste en ejecutar sentencias en la dirección de las flechas y dada una condición que si es verdadera repite el ciclo y si la condición es falsa la repetición termina.

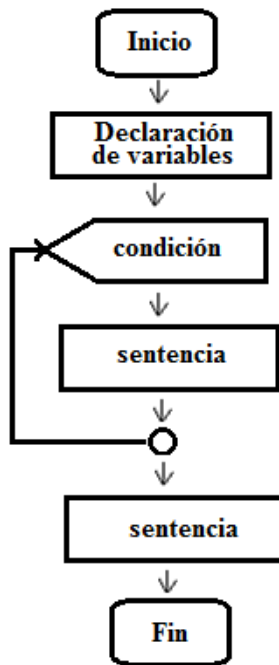


Figura 22. DFD de la estructura MIENTRAS

Ejemplo: En la figura 23 se muestra el DFD del siguiente enunciado: “En un sistema detector de metales instalado en la puerta de ingreso al banco, cuando la puerta está abierta se activa el funcionamiento del dispositivo de detección de metales, y al estar la persona en el radio de acción del dispositivo; si es que detecta un metal en las pertenencias de la persona, la pantalla muestra el mensaje “Prohibido ingresar con objetos autorizados”, en caso contrario “Bienvenido al banco”. El dispositivo funciona hasta que la puerta se cierra”.

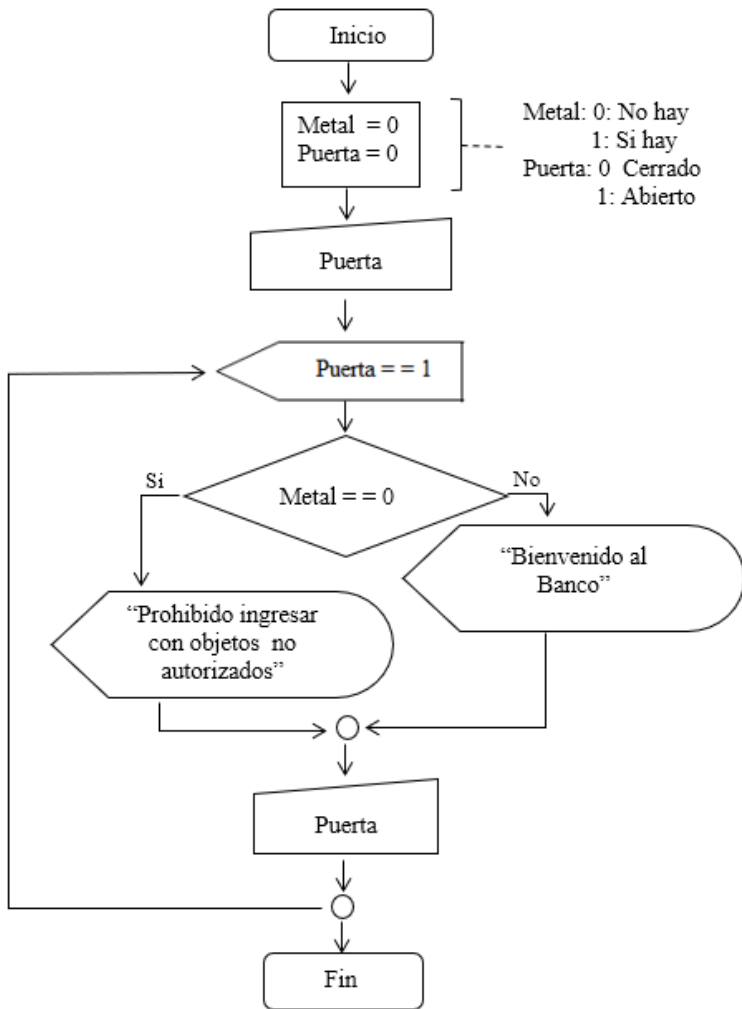


Figura 23. DFD Detección de metal

Ejemplo: En la figura 24 se muestra el DFD del siguiente enunciado: “El programa debe permitir al cliente comprar múltiples productos, al final debe imprimir el total a pagar”.

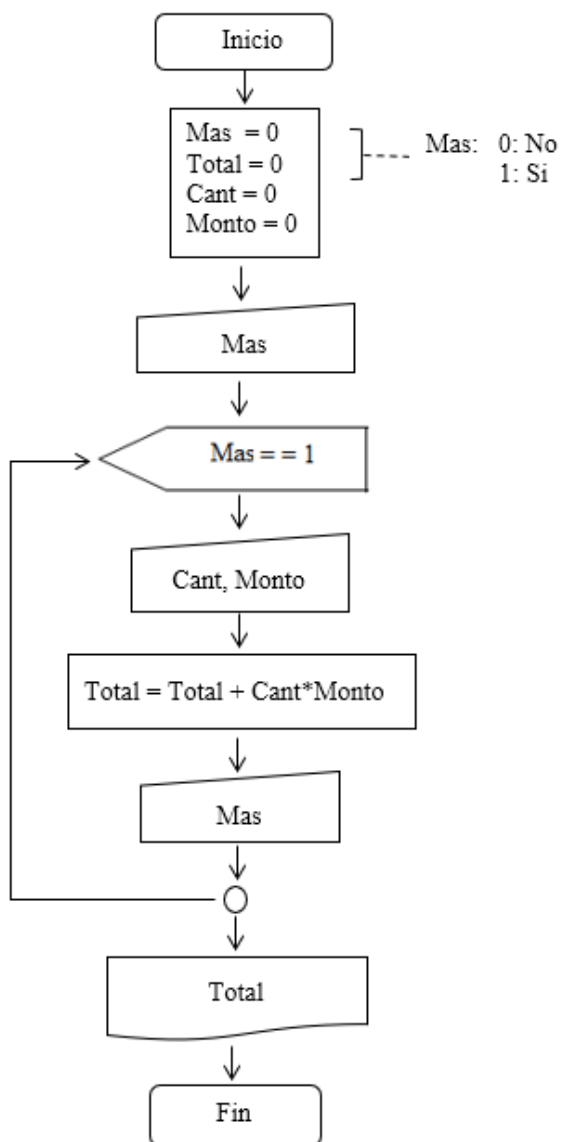


Figura 24. DFD Compras múltiples

5.4.3. Estructura HACER MIENTRAS

En la figura 25, después de la declaración de variables, consiste en ejecutar sentencias en la dirección de las flechas, al menos una vez ejecuta el grupo de sentencias y dada una condición que si es verdadera repite el ciclo y si la condición es falsa la repetición termina.

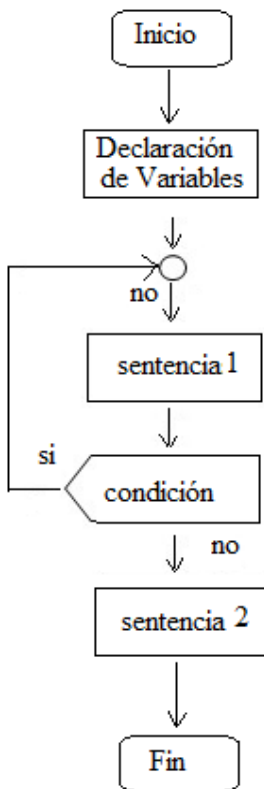


Figura 25. DFD de la estructura HACER MIENTRAS

Ejemplo: En la figura 26 se muestra el DFD del siguiente enunciado: “Control de stock. Ingresar e imprimir los productos (denominación, cantidad) existentes en almacén”.

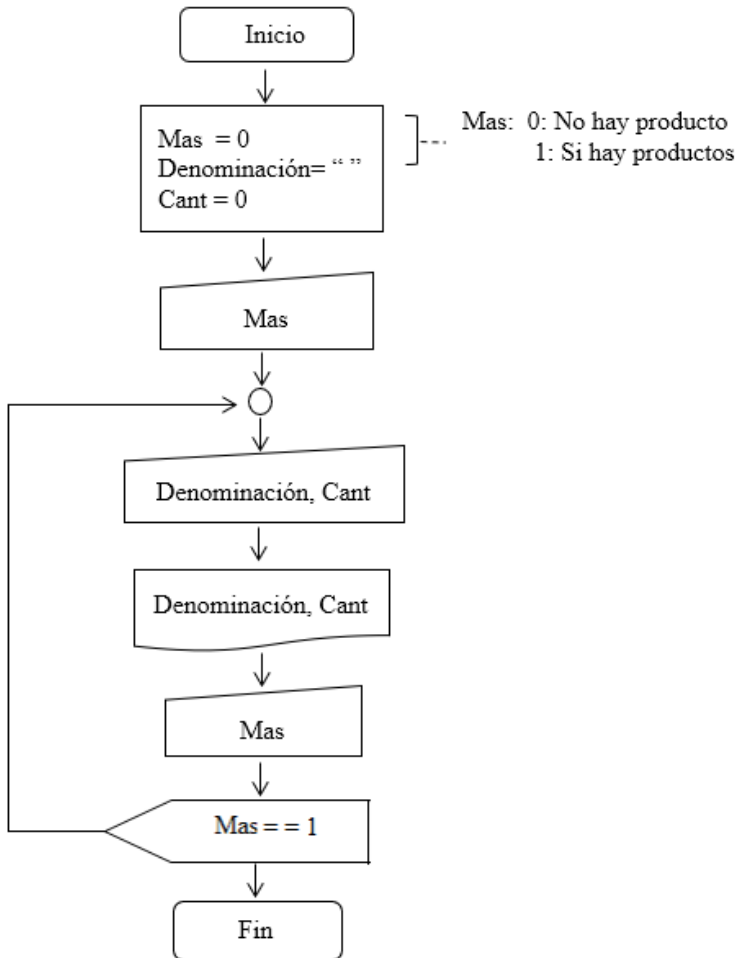


Figura 26. DFD Control de stock

Ejemplo: En la figura 27 se muestra el DFD del siguiente enunciado: “Se necesita saber cuántos sitios arqueológicos existen en el Perú. Mostrar la denominación y lugar de los sitios arqueológicos”.

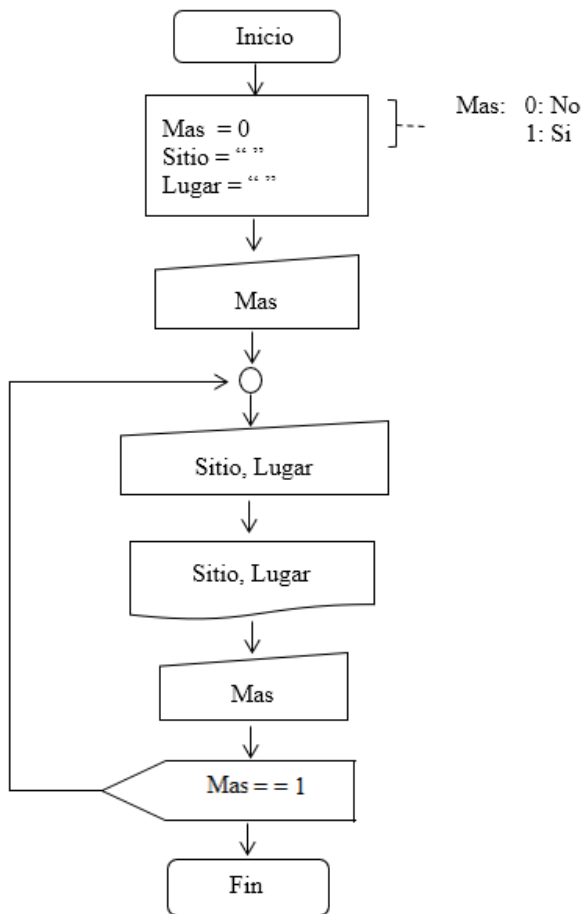


Figura 27. DFD Control de sitios arqueológicos

Preguntas de repaso

Elaborar el DFD de los siguientes enunciados:

1. “Por cada aporte se emite una boleta con el total aportado y la fecha actual”.
2. “Todas las mujeres reciben la donación de víveres de la Oficina de Bienestar Social”.
3. “En el restaurante se publica el menú del día: nombre del plato y precio”.
4. “En la vitrina más grande de la tienda se debe publicar el nombre y precio de los artículos en venta”.
5. “La atención al cliente en tienda es por orden de llegada y para ello se emite un ticket de control de atención; la misma que se le entrega al cliente”.
6. “En la lista de asistencia a clase se registra el nombre (s), apellidos y firma del alumno, por cada vez que asiste a clase”.
7. “Todos los asistentes al cine cuentan con su boleta de entrada que contiene el número de sala, nombre de la película, fecha y hora”.
8. “Ingresar y mostrar la denominación y cantidad por producto existente en el almacén”.
9. “En los días no laborables, el control de asistencia se realiza previa autorización del jefe inmediato del personal de la empresa, por lo que el sistema de control biométrico, debe

- imprimir el reporte de ingreso con el nombre del empleado, la fecha y hora de lectura de huella digital”.
10. “Mostrar por pantalla las siguientes opciones: 1.- Nuevo 2.- Abrir 3.- Grabar 4.- Cerrar”.
 11. “La Tienda cierra media hora antes del horario normal de trabajo diario, momento en que el vendedor ingresa los montos vendidos. El programa informático debe calcular el total vendido diariamente”.
 12. “En las Olimpiadas Estudiantiles solo pueden participar los alumnos matriculados. Se realizan las inscripciones y por cada participante se imprime el DNI, Nombres y apellidos, y la disciplina deportiva a la que participará”.
 13. “Los Ingenieros tienen que tener el grado de maestría para enseñar en la Universidad Nacional de San Martín – Tarapoto”.
 14. “La promoción de helados de cinco bolas es la de mayor demanda en la heladería, indistintamente del sabor, al gusto del cliente. Por cada venta se emite el comprobante de pago y el agradecimiento por su compra”.
 15. “Del total de alumnos matriculados en la asignatura de algoritmos. ¿Cuántos son varones y cuántos son mujeres?”.
 16. “Del total de celulares de la vitrina. ¿Cuántas de ellas son de la marca Samsung?”.

17. “En un sistema de control biométrico, cada vez que lee la huella digital corresponde al registro de asistencia. Entonces, por cada lectura mostrar en pantalla el mensaje “Gracias” si es que existe la huella en el sistema y el mensaje “Volver a intentar” si es que no existe la huella en el sistema”.
18. “Durante el día se vende un promedio de 20 celulares. Cada celular está asociada a un Plan de servicio. ¿Cuántas celulares de la venta diaria corresponden al Plan A y cuántas corresponden al Plan B?”.
19. “Si un vendedor no logra vender 15 motos lineales al mes, es despedido. El control de ventas se hace a fin de mes. Reportar cuántos vendedores han sido despedido de la empresa”.
20. “Hoy lunes, la Bodega de la esquina tiene para la venta productos registrados por nombre, cantidad, unidad de medida, precio unitario de compra y precio unitario de venta. Al registrar el producto, el programa debe validar que el precio unitario de venta debe ser mayor que el precio unitario de compra y la cantidad mayor a cero”.
21. “Para usar un sistema informático, el usuario debe ser autorizado por un *login* y *password* (asignar un valor a cada uno); por lo que, cada vez que ingrese el usuario debe verificarse y mostrar por pantalla si es un usuario autorizado”.

22. “El uso de los servicios higiénicos se controla con las palabras “*Open*” y “*Close*”. Para ello las palabras se muestran en la pantalla de un *display* instalado en la parte superior de la puerta. La puerta se activa al empujar desde adentro en “*Open*” y al empujar la puerta desde afuera en “*Close*””.
23. “Antes del examen de admisión los postulantes consultan al sistema informático de admisión ingresando la primera letra de su apellido paterno, y el sistema informático muestra por pantalla en que aula le corresponde dar el examen, según los siguientes criterios:

<u>Primera Letra de Apellido Paterno</u>	<u>Aula</u>
- A hasta la M	1
- N hasta la Q	2
- R hasta la Z	3”

24. “Las mujeres con hijos reciben la donación de víveres de la Oficina de Bienestar Social y las mujeres sin hijos y casadas reciben su vale de consumo por el día de los enamorados”.
25. “El personal técnico instala el sistema operativo en las computadoras de la empresa según el cuadro de asignación funcional del personal asignado a las siguientes áreas: Sistema Operativo *Windows*: Área de ventas, Área de almacén y Área de Comercialización; Sistema Operativo *Linux*: Área de Informática, Área de producción y la Gerencia General. ¿En

- cuántas computadoras se instaló *Windows* y en cuántas computadoras se instaló *Linux*?".
26. "Mostrar por pantalla las siguientes opciones: "1.- Suma 2.- Resta 3.- Multiplicación 4.- División ". Posteriormente debe solicitar dos números, así mismo debe solicitar el número de la operación a realizar sobre los números solicitados previamente. Es decir, si la opción seleccionada es 1 debe realizar la suma de los dos números e imprimir el resultado. Si la opción seleccionada es 2 debe realizar la resta de los dos números e imprimir el resultado. Si la opción seleccionada es 3 debe realizar la multiplicación de los dos números e imprimir el resultado. Si la opción seleccionada es 4 debe realizar la división de los dos números e imprimir el resultado".
27. "El control de asistencia se hace con un dispositivo de control biométrico, el cual muestra por pantalla los mensajes: 1) Gracias, cuando se realiza entre las 7:00 am y 9:00 am y existe su huella en el sistema de control biométrico; 2) Intente de nuevo, en otras horas del día y no existe su huella en el sistema de control biométrico".
28. "El cliente elige la forma de pago: al contado, con tarjeta o fiado. Por cada uno se aplica la política de venta establecida: al contado, un descuento del 10% del monto total a pagar; con

tarjeta, exactamente el monto a pagar; y fiado, un incremento del 5% del monto total a pagar. Mostrar el monto a pagar”.

29. “Una tienda distribuidora de ropa ha establecido porcentajes de descuento sobre el monto comprado de la siguiente forma:

	Hombres			Mujeres		
Talla	S	M	L	S	M	L
Origen						
Nacional	10%	12%	15%	15%	17%	20%
Importado	5 %.	7%	10%	7%	9%	12%

Imprimir, para un comprador, el monto comprado, el importe del descuento y el monto a pagar”.

30. “Durante la implementación de una red de datos, para la configuración de los equipos, el especialista tiene dos opciones: en el servidor puede instalar *Linux* ó *Windows NT*. Dependiendo de ello, en las estaciones de trabajo instalará: todas *Windows 10*, *Open Office* si es que el servidor tiene *Linux* ó *Office 2016* si es que el servidor tiene *Windows NT*. Imprimir un reporte de configuración de los equipos”.
31. “De las 12 revistas emitidas durante el año 2018, imprimir cuántas son revistas de edición nacional y cuántas son revistas de edición regional”.

32. “Los libros de la biblioteca central son 2000 ejemplares y están ubicadas por materia o temas de contenidos. ¿Cuántos libros existen por cada materia?”.
33. “Por ciclo académico se dictan 34 clases por asignatura. En la lista de asistencia a clase se registra el nombre (s), apellidos y firma del alumno, por cada vez que asiste a clase. Reportar el cuadro de asistencia a clase del alumno Juan Antonio Jiménez Angulo”.
34. “El seguro obligatorio de accidente de tránsito para moto lineal cuesta S/. 100, para auto móviles S/. 85 y para camionetas 4x4 cuesta S/.250. Un punto de venta de la Positiva vende al día 10 unidades y por cada uno gana el 10%. ¿Cuánto gana al día el punto de venta?”
35. “El Servicio de Administración Tributario en Tarapoto, cuenta con 10 tipos de infracciones, cada uno con su respectiva multa. El programa debe imprimir el total recaudado por cada uno de las infracciones (número correlativo, tipo de infracción y total de multa, respectivamente)”.
36. “Un dispositivo de medición de pH en el suelo recolecta valores de una muestra. Mostrar en pantalla la categoría en que se encuentra ese suelo, de acuerdo al siguiente criterio de evaluación:

Categoría	Valor de pH
Fuertemente ácido	< 5.0
Moderadamente ácido	5.1 - 6.5
Neutro	6.6 - 7.3
Medianamente alcalino	7.4 - 8.5
Fuertemente alcalino	8.5”

37. “Mostrar el nombre del grupo ganador del concurso de danzas en la Universidad Nacional de San Martín - Tarapoto, de un total de 20 grupos participantes. El grupo ganador es el que tiene mayor puntaje acumulado, como resultado de la suma de puntos asignado por cada uno de los tres jurados”.
38. “El usuario debe ingresar los siguientes datos de los 7 productos a vender: descripción, cantidad y precio unitario. Al final del día, el programa debe mostrar por pantalla que cantidad se ha vendido y cuando falta vender, por producto, respectivamente”.
39. “Por cada venta de un producto registrado se actualiza la cantidad existente (*stock*). Al final del día, se reporta: las ventas de productos registrados conteniendo número correlativo, nombre del producto, cantidad vendida, unidad de medida, precio de venta), y la lista de productos registrados con menos de 10 cantidades existentes en la bodega conteniendo número

correlativo, nombre del producto, cantidad en stock, unidad de medida, precio de compra”.

40. “La encuesta debe registrarse en el programa considerando. Cada pregunta tiene 3 alternativas, para lo cual se ingresa el número de pregunta y la respuesta marcada por el encuestado. Mostrar el total que marcaron la respuesta a, b y c, por pregunta”.
41. “La empresa operadora de telecomunicaciones registra el tiempo de duración de una llamada, el número de celular de origen y el número de celular destino. Mostrar todas las llamadas realizadas en un día (número correlativo, número de celular origen, número de celular destino y tiempo de duración de la llamada)”.
42. “El programa debe permitir a los alumnos generar su reporte de notas, según el siguiente formato:

UNIVERSIDAD NACIONAL DE SAN MARTÍN		
REPORTE DE NOTAS		
CÓDIGO:71435290 NOMBRES Y APELLIDOS: JUAN PEREZ		
ASIGNATURA	CRÉDITOS	NOTA
Algoritmos	4	10
Matemática Básica	3	12
NUMERO DE ASIGNATURAS APROBADOS		: 2
PORCENTAJE DE ASIGNATURAS APROBADAS		: 50%
NUMERO DE ASIGNATURAS DESAPROBADOS		: 2
PORCENTAJE DE ASIGNATURAS DESAPROBADAS		: 50%
TOTAL DE CRÉDITOS APROBADOS		: 6

43. “Existe tres tipos de usuarios: usuario administrador, usuario operador y el usuario invitado. Cualquiera que sea el usuario debe ingresar con un *login* y *password*; por lo que, cada vez que ingrese el usuario debe verificarse y mostrar por pantalla si es un usuario autorizado y todas las opciones permitidas, respectivamente. El programa debe permitir al usuario autorizado realizar las siguientes acciones:

- El usuario administrador, puede crear nuevos usuarios, imprimir reportes de configuración de la red, eliminar usuarios, hacer copia de seguridad de la base de datos, apagar y prender todo el sistema.
- El usuario operador, puede apagar cualquier dispositivo de la red, verificar el funcionamiento de la red, prender cualquier dispositivo de la red, activar el usuario invitado, instalar todos los sistemas en la red y borrar cualquier sistema instalado en la red.
- El usuario invitado, puede usar los sistemas transaccionales instalados en la red.

Al final debe imprimir un reporte de fecha y hora de inicio y fin de uso de la red por cada usuario”.

44. “El programa debe premiar al cliente por más de S/. 250 de compra. Puede elegir de regalo, uno de los tres licores peruanos ofertados (vino el abuelo, pisco o ron). En el ticket de compra,

además de los licores comprado, imprimir al final de página, el nombre del licor peruano regalado”.

45. “El programa informático debe procesar el movimiento bancario de una cuenta, ingresando inicialmente el número de cuenta (8 dígitos), nombre del propietario y saldo anterior. Luego ingresar todos los movimientos, según el tipo de movimiento (1 = Depósito, 2 = Retiro) y la cantidad que lo represente. Por cada movimiento, actualizar el saldo de la cuenta. Al final, mostrar el número de cuenta, nombre del propietario y saldo actual”.
46. “El programa informático debe controlar diariamente la fabricación de refrescos para atender los pedidos de los clientes. Los pedidos son diversos. Por cada pedido, el total de agua en litros se distribuye según la escala:

<u>Sabor de refresco</u>	<u>Porcentaje de agua</u>
1 = chica morada	10
2 = limonada	15
3 = cocona	20
4 = naranja	25
5 = guanábana	30

47. “El programa informático debe calcular y mostrar el total de botellas de plástico de un litro para cada sabor de refresco, por cada pedido del cliente”.

48. “El Sr. Gustavo contabiliza el total de su fortuna, valorizadas en propiedades 20 millones de dólares, vehículos 25 millones de dólares, 3 millones de dólares en equipos diversos, y 2 millones de dólares en joyas. Decide vender sus bienes y depositar toda su fortuna de la siguiente manera: El 30% en el Banco Continental; el 10% en el Banco de Crédito; el sobrante depositará el 90% en la Cooperativa San Martín y en 2 cooperativas en partes iguales. Mostrar por pantalla cuánto ha depositado en:
- Banco Continental.
 - Banco de Crédito.
 - Cooperativa San Martín.
 - Cooperativa Santo Cristo de Bagazan.
 - Cooperativa la Progresiva.”
49. “Al menos un producto tiene que hacerse vendido durante el día. Mostrar por pantalla, en orden correlativo los productos vendidos (nombre del producto, unidad de medida, cantidad vendida, precio unitario de venta, importe por producto)”.
50. “Del total de asociados, solo entran al sorteo de las 30 canastas por el “Día del Padre” los asociados que están al día en sus aportaciones hasta el mes de mayo. Mostrar los asociados que no están aptos para el sorteo. Realizar el sorteo de la canasta considerando: Primeras 10 canastas se sorteán para los socios

mayores de 70 años. Las canastas sobrantes pasan a formar parte de las canastas a ser sorteadas con los socios que tienen más de 3 años de permanencia en la Cooperativa. Mostrar los ganadores de las 30 canastas”.

51. “La empresa operadora de telecomunicaciones registra el tiempo de duración de una llamada, el número de celular de origen y el número de celular destino. Mostrar todas las llamadas realizadas en un día (número correlativo, número de celular origen, número de celular destino y tiempo de duración de la llamada)”.
52. “El Servicio de Administración Tributaria de Tarapoto, tiene una lista de 10 tipos de infracciones, cada uno con su respectiva multa. El programa debe imprimir la lista de infracciones por día (número correlativo, tipo de infracción y total de multa, respectivamente)”.
53. “El Gobierno Regional de San Martín necesita una lista de beneficiarios de la deuda social del sector educación, para lo cual, el programa debe imprimir la lista considerando: número correlativo, nombres y apellidos del beneficiario y el monto de la deuda, respectivamente”.
54. “Para el concurso de danzas organizado por la Universidad Nacional de San Martín, se están distribuyendo los tickets de entrada a S/. 5.00. El programa debe imprimir el total de

entradas distribuidas entre los alumnos y docentes, hasta la fecha; así como el total de tickets pagados”.

55. “La tienda, por el “Día de la Madre”, oferta sus productos para mujeres. Descuento del 10% por la compra de un monto total entre 50 y 80 soles. Descuento del 15% por la compra de un monto total pasado los 80 hasta 100 soles. Descuento del 20% por la compra de un monto total pasado los 100 soles. Mostrar el monto total vendido aplicando la oferta de sus productos para mujeres, respectivamente”.
56. “En una tienda, la venta de sus productos directamente al cliente, se hace de la siguiente manera:
- a) Al cliente le preguntan por el tipo de comprobante de pago que desea, según los formatos 1 y 2.
 - b) Primero se imprime la cabecera del comprobante de pago elegido por el cliente, seguido de las denominaciones de las columnas del contenido, según formato 1 ó 2.
 - c) En el comprobante de pago, el producto solo se imprime una sola vez.
 - d) El importe por producto se calcula multiplicando el precio unitario por la cantidad del producto.
 - e) El importe total es la suma de los importes por producto.
 - f) El IGV es el 18% del importe total.

- g) El sub-total se calcula restando el valor del IGV del importe total.
- h) Existen tres formas de pago:
 - (1) En efectivo
 - (2) Con tarjeta
 - (3) Ambos (con tarjeta y en efectivo)
- i) Para el caso de pagar en efectivo se debe mostrar en pantalla el vuelto a favor del cliente.

Si el cliente concluye pagando el importe total de lo vendido, se le agradece por su compra; en caso contrario se anula el comprobante de pago y se le agradece por su visita”.

Rescapitulación del arte

Para el diseño del arte se requiere disciplina y secuencia lógica, utilizando los símbolos que permiten la codificación de las instrucciones que construyen la respuesta computacional al enunciado planteado. Partimos de una forma estructurada la secuencia de instrucciones; luego se responde a los controles de decisión simple y múltiple, y dependiendo de las repeticiones de algunas instrucciones con parámetros o argumentos se utiliza las estructuras de repeticiones PARA, MIENTRAS o HACER MIENTRAS. Por lo que, la creatividad del programador marca la diferencia entre las versiones del diseño del arte para programar computadoras.

CAPÍTULO VI PROGRAMACIÓN DEL ARTE

La gran finalidad de la vida no es el conocimiento, sino la acción.

Thomas Henry Huxley

En la actualidad existen diferentes lenguajes de programación para crear programas que pongan en práctica el arte y están agrupados básicamente en dos tipos:

- Lenguajes de programación de alto nivel, que son parecido al texto de un lenguaje natural, inglés, español, etc.
- Lenguajes de programación de bajo nivel, cuya sintaxis se asemeja más a lo que es el código máquina del computador, es decir los unos y los ceros.

En el presente libro usamos el lenguaje de programación *Python* como una forma introductoria a la programación de computadora, que a la fecha cuenta con la versión 3.7.2 (23 de diciembre del 2018), que es un lenguaje de programación de alto nivel. Por lo que se pretende explicar varias características del lenguaje y del sistema *Python*, así como la mecánica de uso del interprete, a través de ejemplos.

6.1. El lenguaje de programación Python

Según Python (2019).

Python es un lenguaje de programación muy potente. Se usan para muchas aplicaciones, en algunas universidades como lenguaje de programación introductorio, porque es fácil de aprender; también lo usan desarrolladores o programadores profesionales en lugares como *google*, la NASA, etc.

Python es un lenguaje de programación de tan alto nivel que algunos dicen que si sabes inglés ya sabes programar en *Python*.

Python fue creado por el Holandés Guido Van Rossum, año 1990.

Presentamos algunas características debido a que cuanto más usamos *Python* aumentan sus particularidades:

- Tiene un entorno de desarrollo amigable y adaptable a cualquier IDE, que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de *software*.
- Tiene una gramática sencilla, clara y muy legible, lo que le permite prescindir de todos aquellos símbolos y particularidades que tienen otros lenguajes de programación. Por ejemplo, prescinde de los puntos y comas al final de cada sentencia. A diferencia de otros lenguajes que obligan poner

símbolos como puntos y comas al final de una sentencia o instrucciones.

- Es *open source* o código abierto.
- Es de naturaleza interpretada. *Python* es un lenguaje interpretado, no es necesario compilar ni enlazar, esto permite ahorrar mucho tiempo durante el desarrollo.
- Es extensible, fácil de agregar una nueva función o módulo al intérprete.
- Es un lenguaje de programación dinámicamente tipado. Esto significa que una misma variable puede tomar valores de distinto tipo en distintos momentos. Quiere decir que el lenguaje de programación distingue de una manera muy clara una variable del tipo entero de una variable de tipo *string*. Todo esto ocurre durante la ejecución del programa. Otros lenguajes de programación son de tipado estático por que las variables se declaran en el programa antes de ejecutarlo.
- Tiene módulos y una librería estándar que viene por defecto, en cualquier versión instalada en su computador.
- Es un lenguaje de programación orientado a objetos.
- Es versátil, nos sirve para crear aplicaciones o programas de escritorio, servidor y web.
- Es multiplataforma, funciona de la misma manera en *Windows*, *Unix* y *macOS*.

6.2. Entorno de desarrollo del lenguaje de programación Python

Una vez instalado el paquete de la versión de *Python* que Usted eligió y para interactuar dinámicamente con el interpretador de *Python* se puede utilizar el IDLE u otro IDE de libre disponibilidad.

El IDLE de *Python* que viene en el paquete del instalador tiene dos ventanas: la ventana *Shell* de *Python* (intérprete interactivo) con colores de entrada de código, salida y mensajes de error, y la ventana del editor; pudiendo tener múltiples ventanas de editor simultáneamente.

En la figura 28 se muestra la ventana *Shell* desde la cual podemos programar. Cuenta con una entrada de tres símbolos de mayor, llamado *prompt* interactivo, es la marca del punto exacto desde la consola dentro del IDLE donde introduciremos nuestras instrucciones *Python*; también llamado línea de interprete. La línea de interprete que no tiene el *prompt* interactivo corresponden a la línea de salida.


```
Python 3.7.2
File Edit Debug Options Help
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC
v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more
information.
>>>
```

Figura 28. *Prompt* de la ventana *Shell*

Como *Python* es un lenguaje interpretado (interprete interactivo), se comporta como por ejemplo lo hace una calculadora de escritorio, práctica; así como probar funciones y escribir programas descartables.

En la figura 29 se muestra un ejemplo del uso de la ventana *Shell*. En pantalla observamos el mensaje “Bienvenido al lenguaje de programación Python”, donde la instrucción está en una línea.

```
>>> print("Bienvenido al lenguaje de programación
Python")
Bienvenido al lenguaje de programación Python
>>>
```

Figura 29. Ejemplo de uso de la ventana *Shell*.

Si se quiere introducir más instrucciones en esta línea de código, se usa el separador de instrucciones punto y coma; en la figura 30 se muestra un ejemplo: mostrar en consola los mensajes “Hola alumnos” y “Saludos a todos”.

```
>>> print("Hola alumnos"); print("Saludos a todos")
Hola alumnos
Saludos a todos
>>>
```

Figura 30. Ejemplo con más de una instrucción por línea en la ventana *Shell*

Cada vez de terminada de introducir la instrucción al final de la línea de código, se presiona *Enter*. Pero para el caso de la codificación de un bucle al termino del bucle presiona dos veces *Enter*.

Por otro lado, los comentarios se introducen utilizando el símbolo numeral # o el carácter hash, tal como se muestra en la figura 31, y no son interpretados por *Python*. Los comentarios son ayuda para el programador dentro del código del programa.

```
>>> #Instrucciones de mensajes
>>> print("Me llamo Julio")
Me llamo Julio
>>>
```

Figura 31. Ejemplo de instrucción de comentarios

Otro uso del numeral o hash es para después de tener una cierta cantidad de líneas de código, se desea hacer prueba de parte de ella, y las instrucciones que no participan en la prueba son comentadas.

También los comentarios se usan para instruir o dar a conocer a otro programador de lo que hace esa instrucción, o el código programado.

Existe en *Python* la posibilidad de separar varias líneas de código usando el símbolo barra invertida `\`, consiguiendo que una misma instrucción este dividida en varias líneas. Es un salto de línea con la misma instrucción. En la figura 32 se muestra el ejemplo: Declaramos una variable `mi_profesión` y mostrar el valor de la variable. Como observará la palabra Marcos está pegado a la palabra Perú, falto dejar un espacio en blanco después de Marcos o introducir un espacio en blanco antes de la palabra Perú.

```
>>> mi_profesión="Licenciado en computación \  
de la Universidad Nacional Mayor de San Marcos \  
Perú"  
>>> mi_profesión  
'Licenciado en computación de la Universidad Nacional  
Mayor de San MarcosPerú'  
>>>
```

Figura 32. Ejemplo de uso de barra invertida

En el caso de elaborar un bloque de código con un sin número de instrucciones, en *Python* se puede hacer una indentación o una sangría que es como una tabulación para identificar la pertenencia de una línea de código al bloque general. En la figura 33 se muestra un ejemplo: hacer un bucle para mostrar los primeros

n números, donde n toma el valor cinco; observamos dos líneas de código dentro del bucle.

```
>>> n=0
>>> for i in range (5) :
    n+=1
    print (n)

1
2
3
4
5
>>>
```

Figura 33. Ejemplo de uso de una indentación

Se dice que estamos usando un intérprete en modo interactivo cuando los comandos son leídos desde un terminal. Es decir, desde un *prompt* primario, luego va esperando un *prompt* secundario, y así sucesivamente.

Si queremos guardar las instrucciones que se van creando, podemos hacerlo con la opción *File* y *Save As*. Cuando deseamos usar este archivo, desde la opción *File* y *Open* levantamos el archivo antes guardado.

El *Shell* interactivo funciona como un editor de textos, donde podemos escribir programas o script en el intérprete de comandos o archivos en lotes.

Python permite separar el programa en módulos que pueden reusarse en otros programas. Algunos de estos módulos proveen la opción de trabajar con entrada/salida de archivos, llamadas al sistema, *sockets*, e interfaces a sistemas de interfaz gráfica de usuario como Tk.

6.3. Sintaxis básica del lenguaje de programación Python

Tipos de datos

Python básicamente considera tres tipos de datos: Numéricos, Textos y Booleanos, como se muestra en la figura 34.

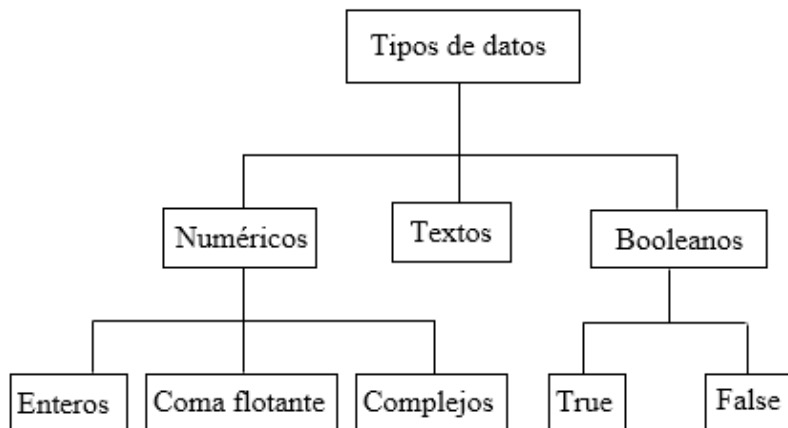


Figura 34. Tipos de datos en *Python*

En los datos tipo numéricos están los números enteros (*int*), los números decimales o de coma flotante (*float*) y los números complejos (usados para cálculos matemáticos). Los números son

interpretados por *Python* introduciendo el número o través de la variable que contiene el número.

Los datos tipo texto pueden ser simples o dobles o comilla triple.

Los datos tipo booleanos pueden ser true o false.

Operadores

Python trabaja con los operadores: operadores aritméticos, operadores de comparación, operadores lógicos, operadores de asignación y operadores especiales, tal como se muestra en la figura 35.

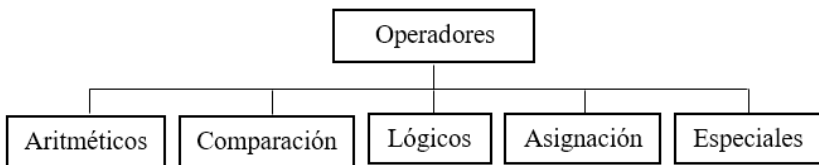


Figura 35. Operadores en *Python*

En cuanto a los operadores aritméticos, *Python* interpreta las siguientes operaciones, tal como se muestra en la figura 36: suma (+), resta (-), multiplicación (*), división (/), exponente (**), división entera (//) y módulo (%), que es el resto de una división. *Python* trabaja como una calculadora.

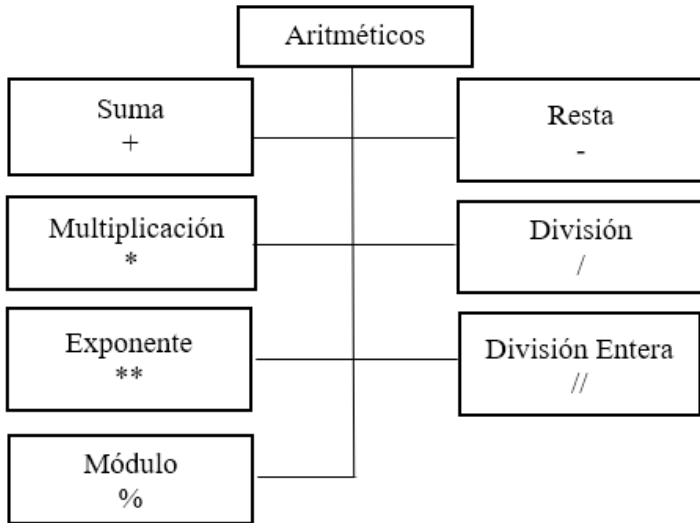


Figura 36. Operadores aritméticos

En la figura 37 se muestra un ejemplo de operadores aritméticos en *Python*.

```
>>> #suma
>>> 3 + 4
7
>>> #resta
>>> 7 - 2
5
>>> #multiplicación
>>> 3 * 2
6
>>> #división
>>> 30 / 5
6.0
>>> #exponente
>>> 8 ** 2
64
```

```
>>> #división entera
>>> 7 // 2
3
>>> #módulo
>>> 9 % 3
0
>>>
```

Figura 37. Ejemplo de operadores aritméticos

Python interpreta los siguientes operadores de comparación, tal como se muestra en la figura 38: Igual que (`==`), diferente que (`!=`), mayor que (`>`), menor que (`<`), mayor o igual que (`>=`) y menor o igual que (`<=`).

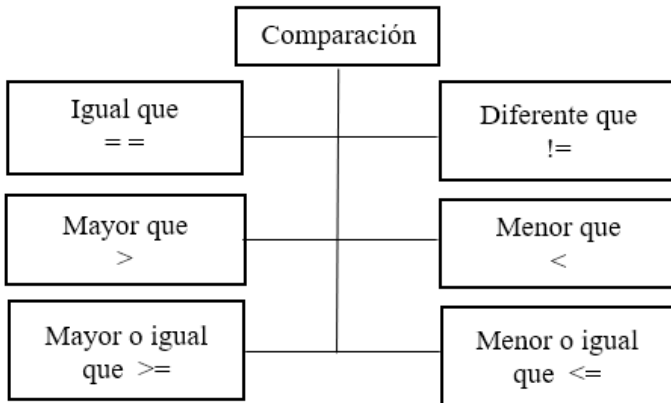


Figura 38. Operadores de comparación en Python

En la figura 39 se muestra un ejemplo de uso del operador igual que.


```
>>> 9 == 10
False
>>> 9 == 9
True
>>>
```

Figura 39. Ejemplo operador igual que
En la figura 40 se muestra un ejemplo de uso del operador diferente que.

```
>>> 10 != 4
True
>>> 10 != 10
False
>>>
```

Figura 40. Ejemplo operador diferente que
En la figura 41 se muestra un ejemplo de uso del operador mayor que.

```
>>> 9 > 12
False
>>> 12 > 9
True
>>>
```

Figura 41. Ejemplo operador mayor que

En la figura 42 se muestra un ejemplo de uso del operador menor que.

```
>>> 7 < 3
False
>>> 3 < 7
True
>>>
```

Figura 42. Ejemplo operador menor que

En la figura 43 se muestra un ejemplo de uso del operador mayor o igual que.

```
>>> 8 >= 8
True
>>> 8 >= 7
True
>>>
```

Figura 43. Ejemplo operador mayor o igual que

En la figura 44 se muestra un ejemplo de uso del operador menor o igual que.

```
>>> 9 <= 7
False
>>> 7 <= 7
True
>>> 7 <= 9
True
>>>
```

Figura 44. Ejemplo operador menor o igual que

Python interpreta los siguientes operadores lógicos, tal como se muestra en la figura 45: AND, OR y NOT.

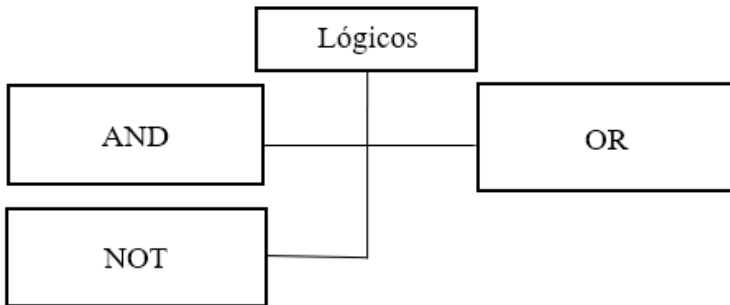


Figura 45. Operadores Lógicos de *Python*.

Python interpreta los siguientes operadores de asignación, tal como se muestra en la figura 46: igual (=), incremento (+=), decremento (-=), multiplicación binaria (*=), división binaria (/=), módulo binario (%=), potencia binaria (**=) y decisión entera (//=).

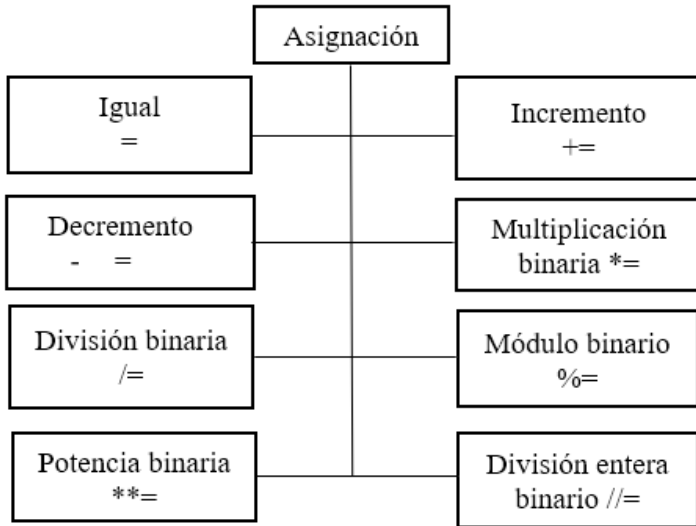


Figura 46. Operadores de asignación en *Python*

Python interpreta los siguientes operadores especiales, tal como se muestra en la figura 47: IS, IS NOT, IN y NOT IN.

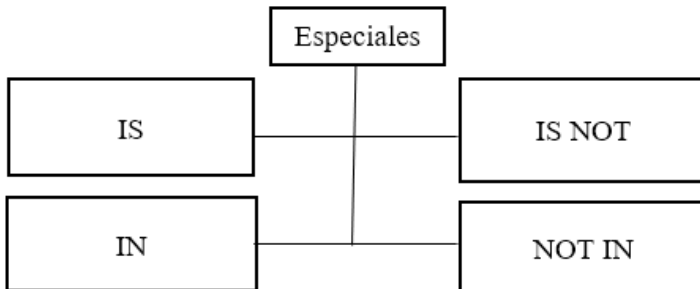


Figura 47. Operadores especiales en *Python*

Variables

Una variable es un espacio de la memoria del computador donde se almacenará un valor que podrá cambiar durante la ejecución del programa.

Por ejemplo, en la figura 48 la variable en la memoria del computador está representado como un espacio físico (memoria RAM) asignado el valor 10, y durante la ejecución del programa puede cambiar ese valor.

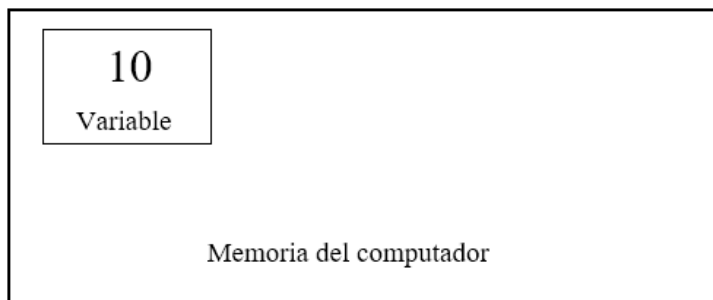


Figura 48. Variable en la memoria del computador

En el *Shell* interactivo de *Python*, el signo igual (=) como ya vimos anteriormente es un operador de asignación y se usa para asignar un valor a una variable. Después de ello, ningún resultado es mostrado antes del próximo *prompt*, tal como se muestra en la figura 49.

```
>>> monto = 99.90
>>> cantidad = 12
>>> monto * cantidad
1198.8000000000002
>>>
```

Figura 49. Asignación de valor a una variable

También, el nombre de una variable puede ser una palabra o la primera letra de una palabra, adicionalmente puede componer palabras o letras separadas por el carácter subrayado, tal como se muestra en la figura 50.

```
>>> mi_nombre = "Julio César"
>>> apellidos="Rodríguez Sigüeñas"
>>> mi_nombre + apellidos
'Julio César Rodríguez Sigüeñas'
>>>
```

Figura 50. Variables con nombres de formas diferentes

Si una variable no está definida o declarada con un valor asignado, al intentar usarla generará error, tal como se muestra en la figura 51.

```
>>> valor
Traceback (most recent call last):
  File "<pyshell#62>", line 1, in <module>
    valor
NameError: name 'valor' is not defined
>>>
```

Figura 51. Error de variable no declarada

Respecto al tipo de variable, no lo establece el contenedor, si no el contenido; es decir si se declara una variable asignando el valor 5, esta variable es de tipo entera. Otro ejemplo, si se declara una variable asignándole el valor “Julio”, entonces se trata de una variable del tipo texto.

En *Python* absolutamente todo es un objeto; entonces las variables son objetos. No olvidar que *Python* es un lenguaje de programación orientado a objeto.

Funciones

Las funciones pueden ser llamadas con argumentos al igual que las palabras claves. A continuación, presentamos las funciones más usadas de *Python*.

Las funciones de *Python* a usar son denominadas funciones predefinidas.

Type(). La función predefinida `type()`, me devuelve el tipo de variable y lo hace con la nomenclatura *class*. Observemos en la figura 52, los resultados mostrados se antepone la palabra *class*, quiere decir que la variable nombre está considerado como una clase.

```
>>> nombre = 15
>>> type(nombre)
<class 'int'>
>>> nombre = "Laura"
>>> type(nombre)
<class 'str'>
>>> nombre = 12.45
>>> type(nombre)
<class 'float'>
>>>
```

Figura 52. Ejemplo de la función predefinida *type*

Print. La función `print()` muestra el contenido o valores que están dentro del paréntesis. En la figura 53, observemos los resultados de la función `print(mensaje)` para la variable `mensaje` que tiene el valor entre tres comillas. También, la función `print()` sin valor dentro del paréntesis.

```
>>> mensaje = """"Estamos avanzando
seguiremos avanzando
hasta la próxima """"
>>> print(mensaje)
Estamos avanzando
seguiremos avanzando
hasta la próxima
>>>
```

Figura 53. Ejemplo de la función *print*

Range. La función *range* (rango en el idioma español) tiene argumentos de inicio (*start*) y pare (*stop*), como se muestra en la figura al introducir el nombre de la función y al abrir

el paréntesis de apertura inmediatamente aparece el mensaje de ayuda interactiva. Esto siempre va ocurrir con todas las funciones a usar por el *prompt*, tal como se muestra en la figura 54.

```
>>> range(range(stop) → range object
          range(start, stop[, step]) → range object
```

Figura 54. Función *range*

En la figura 55 se muestra una aplicación de la función *range* con un valor de argumento y otro con dos argumentos. Como se observa en el caso de un argumento automáticamente asigna el valor cero al argumento *start*; es decir acepta un argumento obligatorio.

```
>>> range(1)
range(0, 1)
>>> range(2,3)
range(2, 3)
>>>
```

Figura 55. Ejemplo de la función *range*

Definición de funciones

Las funciones son un conjunto de líneas de código agrupados (bloque de código) que funcionan como una unidad realizando una tarea específica. Pueden devolver valores. También se les denomina métodos cuando se encuentran definidas dentro de una clase. Las funciones sirven para poder reutilizar el código

cuando sea necesario. Las funciones a crear o definir se denominan funciones propias.

Sintaxis de la definición de una función: La palabra reservada *def* se usa para definir funciones. La línea de instrucciones debe estar debajo de la línea inicial, con sangría. El retorno de un valor es opcional.

1) Def nombre_de_la_función(zona de parámetros):

Instrucciones de la función

Return (opcional)

2) Def nombre_de_la_función():

Instrucciones de la función

Return (opcional)

En la figura 56 se muestra un ejemplo de definición de funciones, presentamos el cálculo del área de un rectángulo, ingresando dos parámetros o argumentos.

```
>>> #Definición de la función cálculo del área
>>> def area (largo, ancho) :
    medida = largo*ancho
    print(medida)

>>> area (10, 14) #usando la función definida
140
>>>
```

Figura 56. Ejemplo de definición de una función definida

Otro ejemplo de definición de funciones se muestra en la figura 57, ingresando dos parámetros o argumentos y retorno de resultado.

```
>>> #Ejemplo con valores de retorno
>>> #Cálculo del área de un rectángulo
>>> def area (largo, ancho) :
    resultado = [ ]
    resultado = largo * ancho
    return resultado

>>> medida = area (10, 14)
>>> medida #Ejecutando con retorno de resultado
140
>>>
```

Figura 57. Ejemplo de función con retorno de resultado

Control de flujos

La aplicación de los operadores de comparación se hace con los condicionales. A continuación, presentamos las sentencias de uso de control de flujos en *Python*.

if. La sentencia condicional *if* se usa para evaluar dos o más condiciones, si se cumple la condición se hace una acción y si no se cumple la condición se hace otra acción. En la figura 58 creamos dos variables asignándole valor, respectivamente, y utilizando un operador de comparación

menor que evaluamos la condición, si se cumple mostramos un mensaje y si no mostramos otro mensaje.

```
>>> variable_uno=17
>>> variable_dos=15
>>> if variable_uno < variable_dos:
    print("No cumple la condición")
else:
    print("Si cumple la condición")

Si cumple la condición
>>>
```

Figura 58. Sentencia if

También, puede haber cero o más bloques *else if* (elif), y el bloque *else* es opcional, como se muestra en la figura 59. Una sentencia *if ... elif ...* sustituye las *switch* o *case* encontrados en otros lenguajes de programación.

```
>>> miembros = int(input("Ingresar el total de miembros de
una familia  "))
Ingresar el total de miembros de una familia  5
>>> mujeres=2
>>> if miembros < 5:
    print("No")
elif miembros == 5:
    varones = miembros - mujeres - 2
    print ("Existen 2 mujeres y 1 varón")
else:
    print("Gracias totales")
Existen 2 mujeres y 1 varón
```

Figura 59. Ejemplo de más de una condición

for. La sentencia *for* en *Python* itera sobre los ítems de cualquier sentencia, en el orden que aparecen en la secuencia. En otros lenguajes de programación como C o Pascal, la sentencia *for* itera sobre una progresión aritmética de números o da la posibilidad de definir el paso final. Un ejemplo de la sentencia *for* en *Python* mostramos en la figura 60, donde se mide el número de letras de las palabras o cadena de caracteres.

```
>>> #Contar el número de letras por palabra
>>> cadena = ["Mujer", "Hermanos", "Familia"]
>>> for p in cadena:
    print(p, len(p))

Mujer 5
Hermanos 8
Familia 7
>>>
```

Figura 60. Ejemplo de la sentencia *for*

Repeticiones o bucles

While Un bucle *while* en *Python* se ejecuta mientras la condición tenga el valor *True* (se cumpla la condición). Cuando la condición tenga el valor de *False*, se termina la ejecución del bucle *while*.

Sintaxis:

While condición

Cuerpo del bucle

En la figura 61 se muestra un ejemplo de uso del *bucle while*.

```
>>> a, b = 0, 1
>>> while b < 1000:
    print(b, end = ', ')
    a, b = b, a + b

1,1,2,3,5,8,13,21,34,55,89,144,233,377,610,987,
>>>
```

Figura 61. Ejemplo de la sentencia *while*

En la figura 62 se muestra un ejemplo del bucle *while* dentro de la definición de funciones, los resultados corresponden a la serie de Fibonacci.

```
>>> #Serie de Fibonacci
>>> def fibonacci(numero): #Escribe hasta número
    a, b = 0, 1
    while a < numero:
        print(a, end=' ')
        a, b = b, a + b
    print()
>>> # Utilizamos la función
>>> fibonacci(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
>>>
```

Figura 62. Ejemplo de definición de una función y bucle *while*

Preguntas de repaso

Programar el arte de los enunciados de las preguntas de repaso del capítulo anterior, utilizando el lenguaje de programación *Python*.

Recapitulación del arte

La programación del arte requiere de un lenguaje de programación de computadoras. Para el presente libro universitario elegimos *Python* por ser un lenguaje muy potente, de fácil aprendizaje y orientado a objeto. En el desarrollo de la sintaxis básica de *Python*, las variables son objetos, las funciones son métodos dentro de una clase.

Anexos

Anexo 1. Otros recursos sobre Python

<https://www.python.org>: El sitio web principal sobre Python. Contiene código, documentación, y referencias a páginas relacionadas con *Python* en la *Web*.

<https://docs.python.org>: Acceso rápido a la documentación de Python.

<https://pypi.python.org>: El índice de paquetes de *Python*, es un listado de módulos de *Python* disponibles para descargar hechos por otros usuarios. Cuando comiences a publicar código, puedes registrarlo aquí así los demás pueden encontrarlo.

<https://code.activestate.com/recipes/langs/python/>: El Recetario de *Python* es una colección de tamaño considerable de ejemplos de código, módulos más grandes, y programas útiles. Las contribuciones particularmente notorias están recolectadas en un libro también titulado Recetario de *Python*.

<https://www.pyvideo.org>: recolecta links de videos relacionados con *Python* de conferencias y reuniones de grupos de usuarios.

<https://scipy.org>: El proyecto “*Python Científico*” incluye módulos para manipulación y cálculo rápido de arreglos además de incluir paquetes para cosas como algebra lineal, transformaciones de Fourier, soluciones no lineales, distribuciones de números al azar, análisis estadísticos y similares.

Glosario de términos

Algoriólogo.- Artista algorítmico.

Algoritmo.- Es el arte de elaborar un conjunto de pasos que representan la solución al problema planteado.

Arte.- Es una forma creativa de expresar la ciencia computacional.

Computadora.- Es cualquier máquina electrónica que solo funciona cuando esta encendido.

Datos de entrada.- Son datos para ser transformados.

Datos de salida.- Son datos que representan a los resultados esperados.

Datos mixtos.- Son datos que pueden estar a la entrada y salida del proceso de transformación.

Enunciado de un problema.- Es un conjunto de palabras que expresan el problema planteado, claros y precisos.

Programa de computadoras.- Son algoritmos expresados en lenguajes compatibles con la computadora.

Recapitulación del arte.- Resumen del capítulo de competencias.

REFERENCIAS BIBLIOGRÁFICAS

- Calderón S. & Ortega J. (2009). *Guías para la elaboración de diagramas de flujo*. Costa Rica
- Guillermo M. (2018). *El arte es un algoritmo*. Recuperado 21:00, octubre 15, 2018 de <https://blogs.upm.es/cuartacultura/2018/03/06/el-arte-es-un-algoritmo-cronica-arco-2018/>
- Joyanes L. (1988). *Fundamentos de programación. Algoritmos y estructura de datos*. España.
- Joyanes L., Rodríguez L., & Fernández M. (1996). *Fundamentos de programación Libro de problemas*. España.
- Pinales F. & Velázquez C. (2014). *Algoritmos resueltos con diagramas de flujo y pseudocódigo*. México.
- Python (2019). *Python*. Recuperado 11:10, enero 03, 2019 de <https://www.python.org/>
- Revista sobredosis (2018). *¿Puede un algoritmo reemplazar a un artista?* Revista cultural. Recuperado 21:00, octubre 15, 2018 de <https://medium.com/revista-sobredosis/puede-un-algoritmo-reemplazar-a-un-artista-5e68dd516778>
- Rosenblatt D. (2018). *Arte: CAN, el algoritmo que produce pinturas y podría reemplazar a los artistas*. Cultura Colectiva. Recuperado de <https://culturacolectiva.com/arte/can-algoritmo-que-produce-pinturas>