

UT1: Desarrollo de software

Diagramas de flujo

EL ALGORITMO

“Método que se realiza paso a paso para solucionar un problema que se termina en un número finito de pasos”

Preciso

- Indicar el orden de realización de cada paso

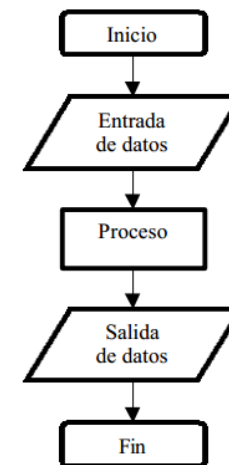
Definido

- Si se sigue el algoritmo dos veces se obtener el mismo resultado cada vez

Finito

- Debe tener un final, un número finito de pasos

Describe tres partes:
ENTRADA,
PROCESO,
SALIDA



EL ALGORITMO

Análisis y
Diseño del
algoritmo



Flujo de operaciones que se han de realizar para su resolución y el orden en que estas operaciones deber ser ejecutadas



Método de
representación



*Diagramas de flujo,
Pseudocódigo,
Diagramas Nassi-Schneiderman (N-S)
Tablas de decisión*



Codificación en un lenguaje
de programación



Ejecución

EL ALGORITMO

Análisis y
Diseño del
algoritmo



Algoritmo para cambiar la llanta a un coche

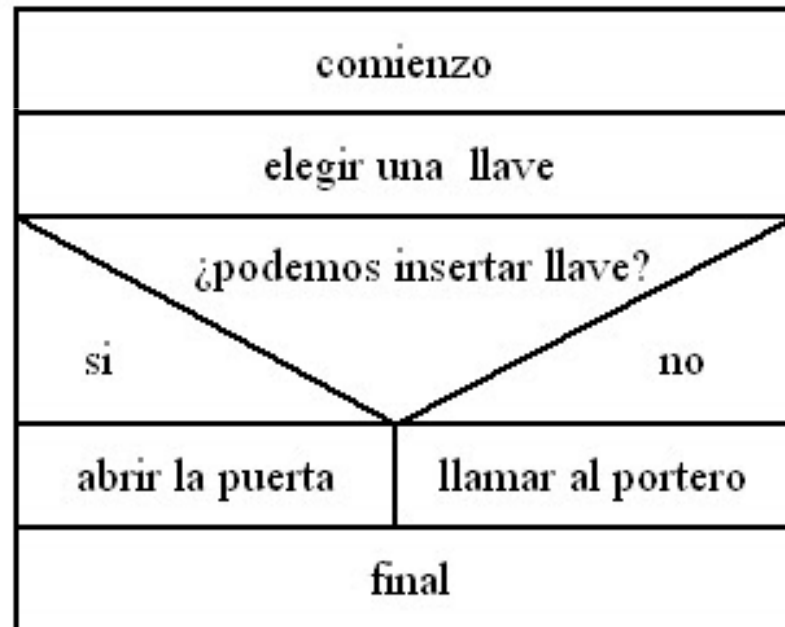
1. Inicio.
2. Traer gato.
3. Aflojar tornillos de las llantas.
4. Levantar el coche con el gato.
5. Sacar los tornillos de las llantas.
6. Quitar la llanta.
7. Poner la llanta de repuesto.
8. Poner los tornillos.
9. Bajar el gato
10. Apretar los tornillos.
11. Fin

EL ALGORITMO

Método de
representación



Diagramas Nassi-Schneiderman (N-S)
Tablas de decisión



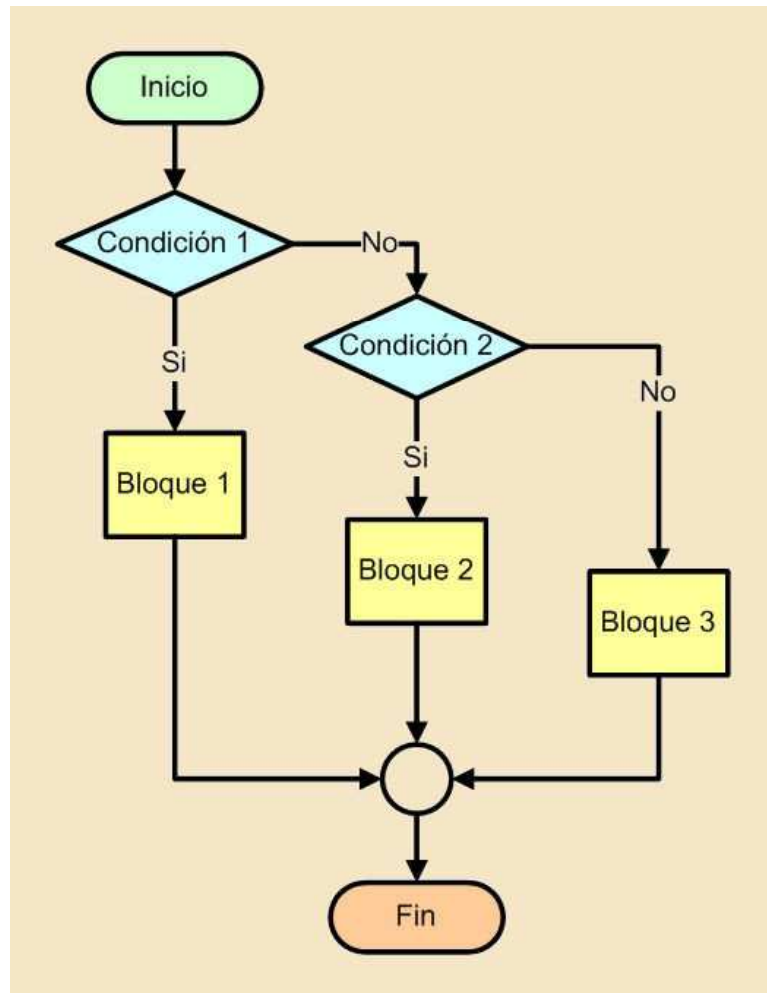
	3	4	6	7	8
Pago con tarjeta oro	Sí	No	Sí	No	No
Pago con tarjeta club	No	Sí	No	Sí	No
Modalidad joven de tarjeta	Sí	Sí	No	No	No
Inviabile					
Descuento 15% (oro)	X		X	$\frac{+}{-}$	
Descuento 5% (club)		X		X	
Descuento 5% (joven)	X	X			
Calcular importe	X	X	X	X	X

EL ALGORITMO

Método de
representación



Diagramas de flujo



- Representaciones **gráficas** de secuencias de pasos a realizar.
- Cada operación se representa mediante un **símbolo normalizado** el Instituto Norteamericano de Normalización (ANSI - American National Standards Institute).
- Las líneas de flujo indican el orden de ejecución.

EL ALGORITMO

Método de
representación


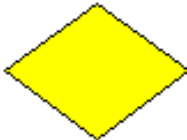






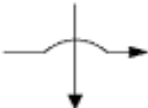



Pseudocódigo

```
1  Proceso pago
2      Escribir "monto consumido";
3      Leer monto;
4      Si monto>50 Entonces
5          |   descuento<-0.2*monto;
6          |   pago<-monto-descuento;
7      Sino
8          |   descuento<-0.1*monto;
9          |   pago<-monto-descuento;
10     FinSi
11     Escribir "total a pagar", pago;
12
13 FinProceso
14
```

- Describen un algoritmo de forma similar a un lenguaje de programación pero todavía cercano lenguaje natural
- **No** está regido por ningún estándar.

SÍMBOLOS

	Inicio/Final Se utiliza para indicar el inicio y el final de un diagrama; de Inicio sólo puede salir una línea de flujo y al final sólo debe llegar una línea		Decisión Indica la comparación de dos datos y dependiendo del resultado lógico (falso o verdadero) se toma la decisión de seguir un camino del diagrama u otro
	Entrada/Salida Entrada/Salida de datos por cualquier dispositivo (scanner, lector de código de barras, micrófono, parlantes, etc.)		Impresora/Documento. Indica la presentación de uno o varios resultados en forma impresa
	Entrada por teclado. Entrada de datos por teclado. Indica que el computador debe esperar a que el usuario teclee un dato que se guardará en una variable o constante		Pantalla Instrucción de presentación de mensajes o resultados en pantalla
	Acción/Proceso Indica una acción o instrucción general que debe realizarse (operaciones aritméticas, asignaciones, etc.)		Conector Interno Indica el enlace de dos partes de un diagrama dentro de la misma página
	Flujo/Flecas de Dirección Indica el seguimiento lógico del diagrama. También indica el sentido de ejecución de las operaciones		Conector Externo Indica el enlace de dos partes de un diagrama en páginas diferentes

ESTRUCTURAS BÁSICAS

Secuencia

Grupo de acciones que se realizan todas y en el orden en que están escritas, sin posibilidad de omitir ninguna de ellas.

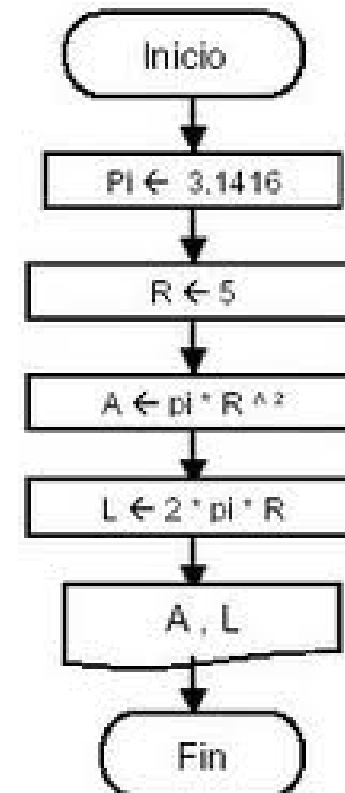
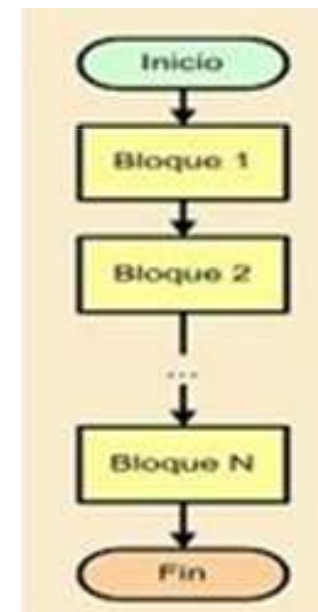
Pseudocódigo:

INICIO

Levante la bocina
Espere tono
Marque el número
Espere que contesten
Hable con la otra persona
Cuelgue la bocina

FIN

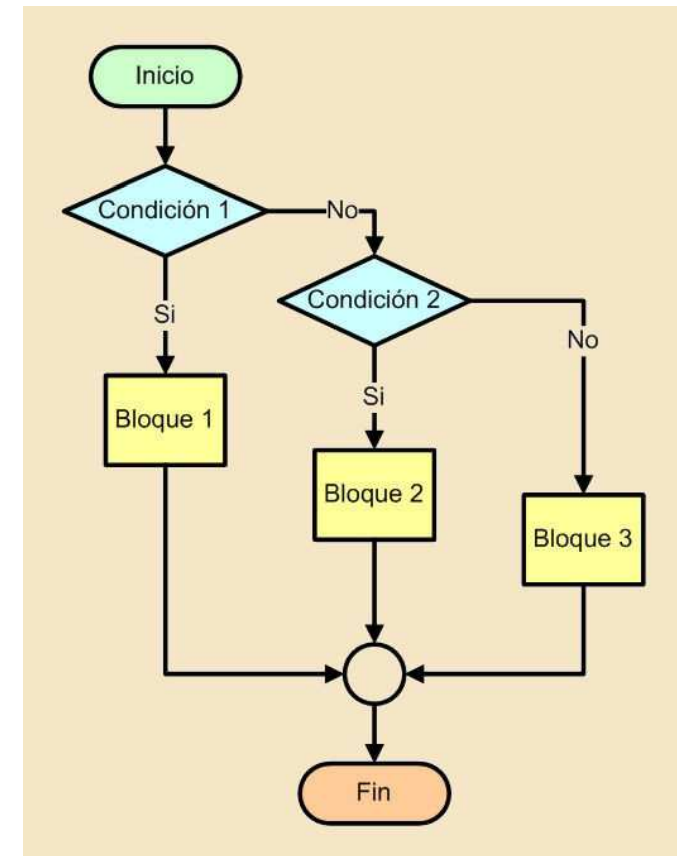
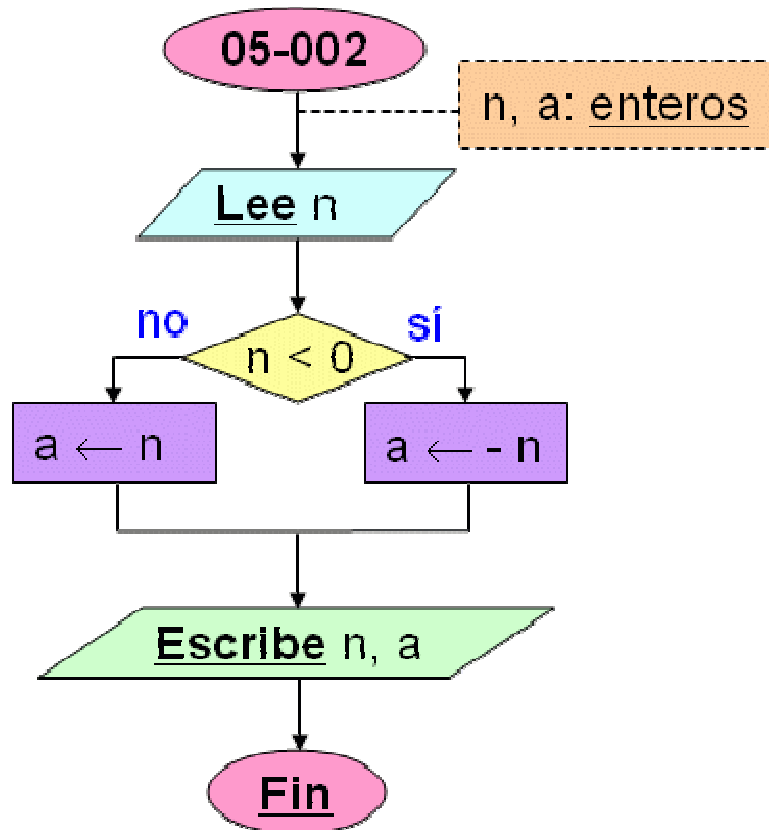
Diagrama de flujos:



ESTRUCTURAS BÁSICAS

Alternativa

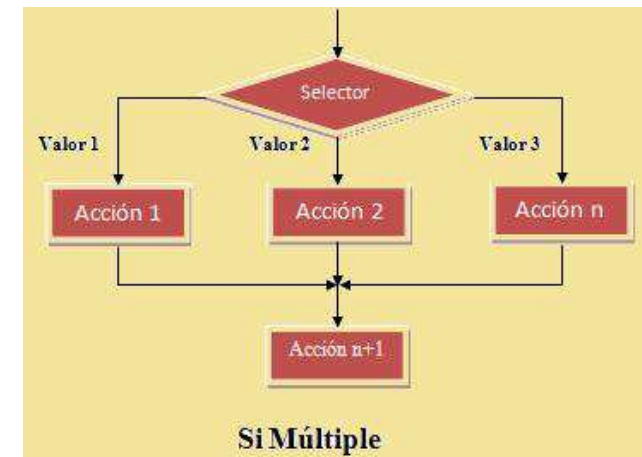
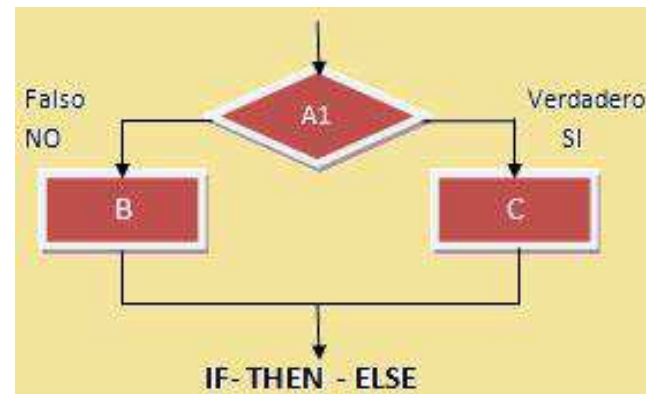
Permite la selección entre dos grupos de acciones dependiendo de que una determinada **condición** se cumpla o no



ESTRUCTURAS BÁSICAS

Alternativa

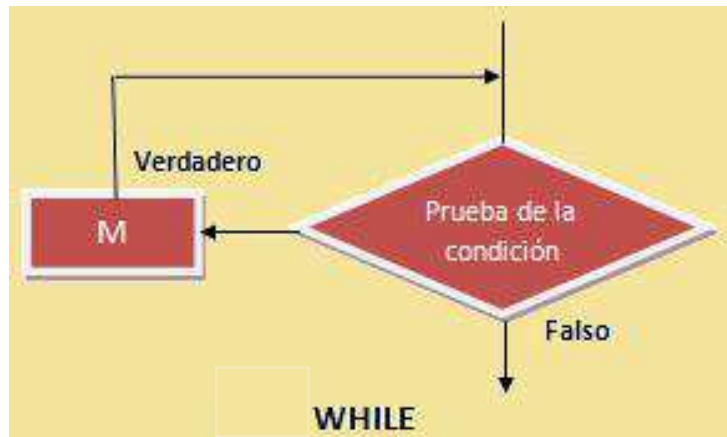
Permite la selección entre dos grupos de acciones dependiendo de que una determinada **condición** se cumpla o no



ESTRUCTURAS BÁSICAS

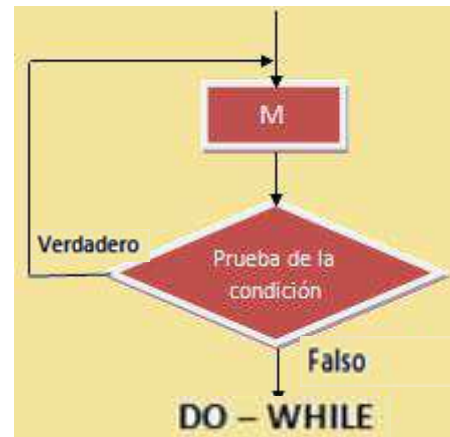
Repetitiva (Iteración)

Permite repetir una o varias instrucciones un número determinado de veces que vendrá determinado por una **condición**



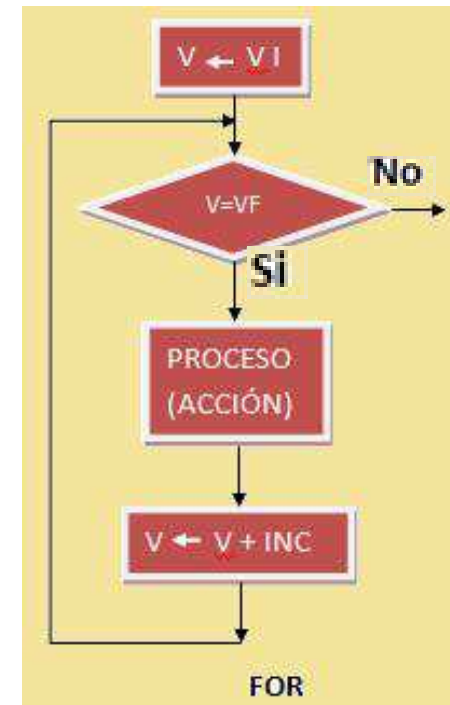
0-N

Termina cuando la condición es **FALSA**



1-N

Termina cuando la condición es **FALSA**



EJEMPLO

Necesitamos hacer un programa que multiplique dos números enteros.

Sabemos que “ $5 \times 3 = 15$ ” es lo mismo que “ $5 + 5 + 5 = 15$ ”.

VARIABLES

<i>Multiplicando</i>	entero (nos indica el número que vamos a sumar)
<i>Multiplicador</i>	entero (nos indica el número de veces que lo vamos a sumar)
<i>Resultado</i>	entero (en esta variable asignaremos el resultado)
<i>Indice</i>	entero (nos indicara el número de veces que el número se ha sumado)

EJEMPLO

Necesitamos hacer un programa que multiplique dos números enteros.

Sabemos que “ $5 \times 3 = 15$ ” es lo mismo que “ $5 + 5 + 5 = 15$ ”.

ALGORITMO

- 1) *Asignamos el número 5 a multiplicando*
- 2) *Asignamos el número 3 a multiplicador*
- 3) *Asignamos el número 0 a resultado*
- 4) *Asignamos el número 0 a indice*
- 5) *Sumamos multiplicando y resultado*
- 6) *Asignamos a resultado la suma*
- 7) *Incrementamos 1 a indice*
- 8) *Mientras indice sea menor a multiplicador regresamos al paso 5 de lo contrario continua*
- 9) *Muestra el resultado*

EJEMPLO

1. Inicio
2. multiplicando = 5
3. multiplicador = 3
4. resultado = 0
5. indice = 0
6. **do**
7. resultado = resultado + multiplicando
8. indice = indice + 1
9. **mientras** indice < multiplicador
10. imprime resultado

