

Introducción a la programación:

Algoritmos

Algoritmo

- La palabra algoritmo proviene del nombre del matemático persa del siglo IX Muhammad ibn Musa al-Jwarizmi
- Un algoritmo es una secuencia ordenada de pasos, sin ambigüedades, que permite la resolución de un problema dado y expresado en lenguaje natural, ej. en castellano.
- Características de un algoritmo
 - Preciso: Indica el orden de realización de cada uno de sus pasos.
 - Finito: si se sigue el algoritmo, se debe terminar en algún momento, es decir tiene un número finito de pasos.
 - Definido: si se sigue dos veces el algoritmo con los mismo datos de entrada, la salida debe ser la misma.

Algoritmo para cambiar una lamparita

- **1. Retirar la lamparita fundida**
 - 1. Colocar una escalera debajo de la lámpara
 - 2. Subir la escalera
 - 3. Desenroscar la lamparita en el sentido contrario de las agujas del reloj
 - 4. Bajar la escalera
- **2. Poner una lamparita nueva**
 - 1. Tomar la lamparita nueva
 - 2. Subir la escalera
 - 3. Enroscar la lamparita en el sentido de las agujas del reloj
 - 4. Bajar la escalera
 - 5. Retirar la escalera

Descripción de algoritmos

- **Metodología para describir algoritmos**
 - **Entrada:** Información dada al algoritmo.
 - **Proceso:** Operaciones o cálculos necesarios para encontrar la solución del problema.
 - **Salida:** Respuestas dadas por el algoritmo o resultados finales de los procesos realizados.
- **El lenguaje algorítmico debe ser independiente de cualquier lenguaje de programación particular, pero fácilmente traducible a cada uno de ellos. Alcanzar estos objetivos conducirá al empleo de métodos normalizados para la representación de algoritmos, tales como los diagramas de flujo o el pseudocódigo.**

Notaciones para describir algoritmos

- **Método informal**
 - **Lenguaje natural**
 - **Ventajas:** comprensible e intuitivo
 - **Inconveniente:** impreciso
- **Métodos formales**
 - **Pseudocódigo**
 - **Lenguaje natural limitado y sin ambigüedad**
 - **Diagramas**
 - **Diagramas de flujo u organigramas**
 - Utiliza un conjunto de símbolos para representar cada estructura de control y mediante líneas de flujo se indica el orden en que se realiza el flujo lógico del algoritmo
 - **Diagramas de Nassi-Schneiderman o Diagramas N-S**
 - Los pasos sucesivos se escriben en cajas con distintas formas según la estructura de control que representen

Notaciones para describir algoritmos

- **Variables**

- **Numéricas:**

- **Enteros:** integer
 - **Reales (punto flotante):** real

- **Alfanuméricas:**

- **Carácter:** char
 - **Cadena de caracteres:** string

- **Lógicas:** boolean

- **Operaciones:**

- **Asignación:** =

- **Aritméticas:** +, -, *, /, ^ (potenciación), MOD (resto división entera)

- **Lógicas:** >, <, =, !=, <=, >=, AND, OR, NOT

- **Trigonométricas:** SIN (), COS (), TAN (), ASIN (), ACOS (), ATAN ()

- **Otros operadores:**

- **Número aleatorio:** RANDOM ()

- **Raíz cuadrada:** SQRT ()

- **Parte entera:** TRUNC ()

- **Concatenar:** +

- **Valor absoluto:** ABS ()

- **Largo de cadena:** LEN ()

Algoritmo que calcule la superficie de un rectángulo proporcionándole su base y altura. (Metodología)

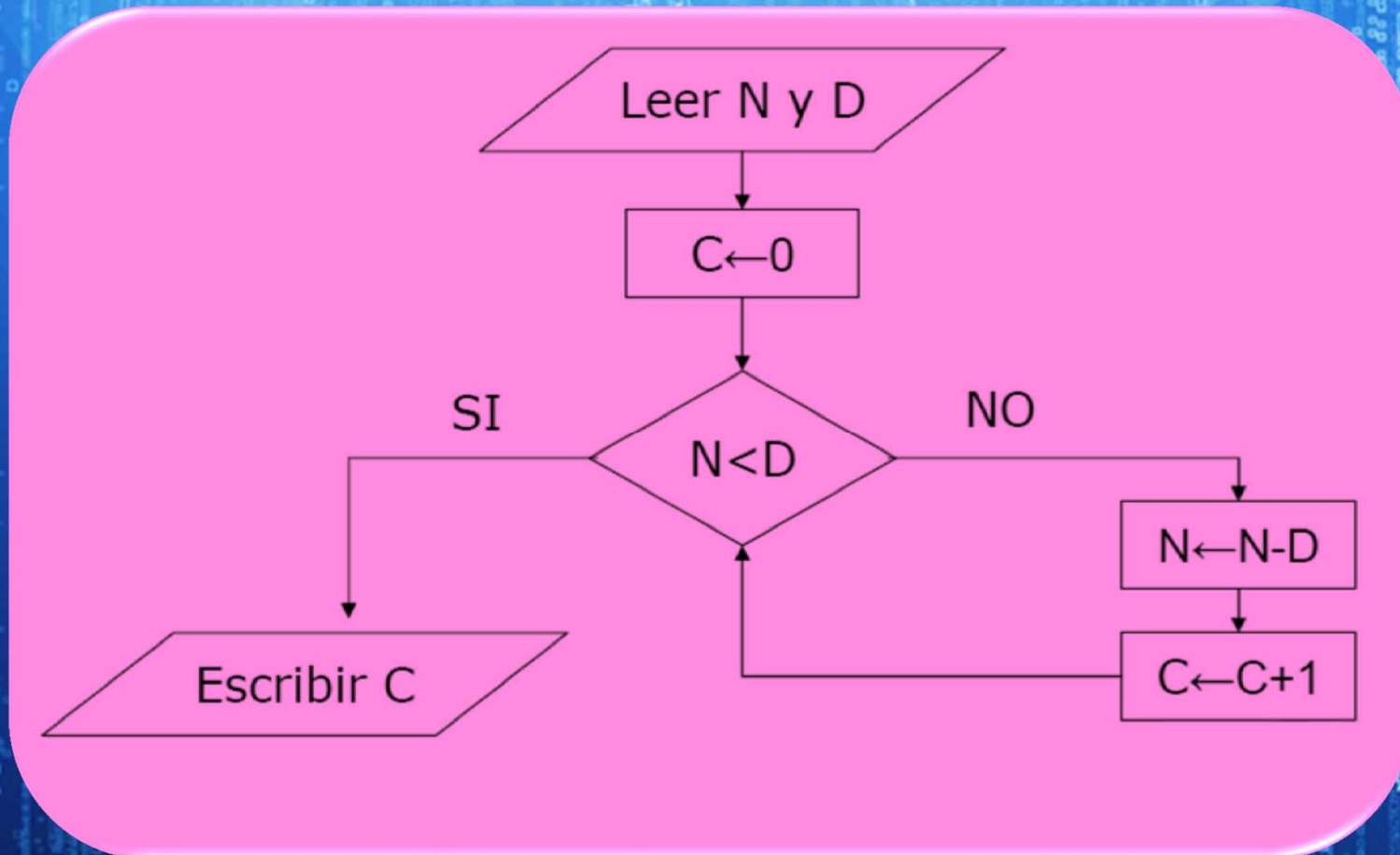
- **Especificaciones de entrada:**
 - ¿Qué datos son de entrada?
 - ¿Cuántos datos se introducirán?
 - ¿Cuántos son datos de entrada validos?
- **Especificaciones de salida:**
 - ¿Cuáles son los datos de salida?
 - ¿Cuántos datos de salida se producirán?
 - ¿Qué formato y precisión tendrán los resultados?
- **El algoritmo que podemos utilizar es el siguiente:**
 - Paso 1: Entrada desde el teclado de los datos de base y altura
 - Paso 2: Calculo de la superficie, multiplicando la base por la altura
 - Paso 3: Salida por pantalla de base, altura y superficie calculada.

Algoritmo para calcular el cociente de una división entera sin usar la operación de división (Lenguaje Natural)

- 1. Leer el numerador N y el denominador D
- 2. Inicializar el cociente a 0 ($C \leftarrow 0$)
- 3. Si $N < D$ ir al paso 7
- 4. Restar a N el valor de D ($N \leftarrow N - D$)
- 5. Incrementar en uno C ($C \leftarrow C + 1$)
- 6. Si $N \geq D$ volver al paso 4
- 7. Mostrar el valor de C



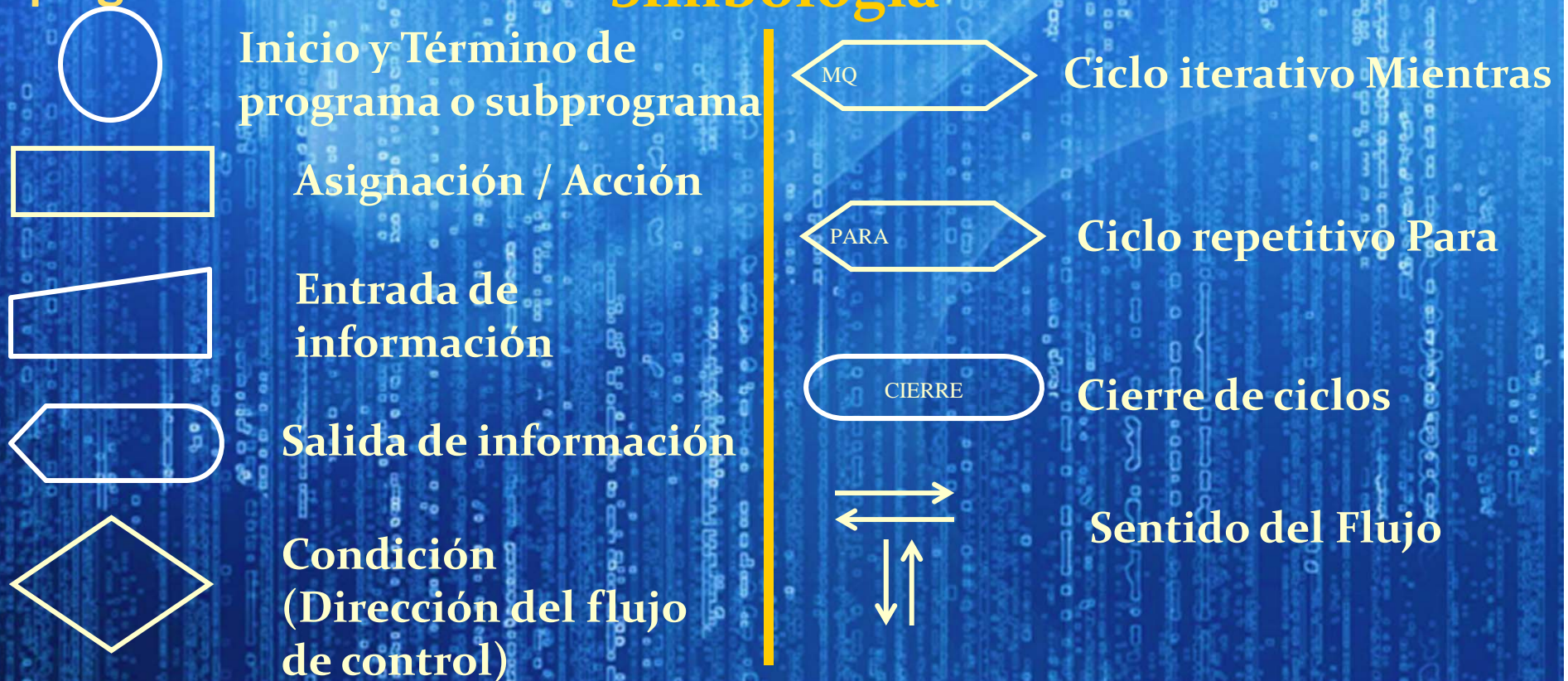
Algoritmo para calcular el cociente de una división entera (Diagrama de flujo)



Diagramas de Flujo

Son gráficas detalladas que representan los pasos que se deben ejecutar dentro de la computadora a fin de producir los resultados esperados. Los Diagramas de Flujos son una técnica estándar para representar algoritmos y son independientes del lenguaje de programación utilizado.

Simbología

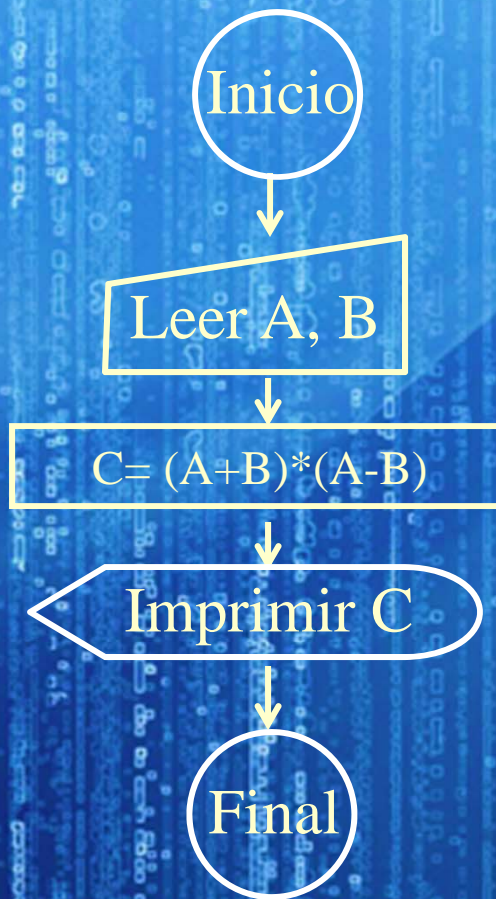


Diagramas de Flujo

- Todo diagrama de flujo debe tener un inicio y un fin.
- Las líneas utilizadas para indicar la dirección del diagrama deben ser rectas, horizontales o verticales, nunca se deben cruzar entre si.
- No deben haber líneas sin conexión a los demás elementos del diagrama de flujo.
- Un diagrama de flujo se debe construir de arriba hacia abajo y de requerirse de izquierda a derecha.
- La notación o símbolos utilizados son independientes del lenguaje de programación utilizado para la elaboración del programa o aplicación.
- No puede llegar mas de una línea de conexión a un símbolo.

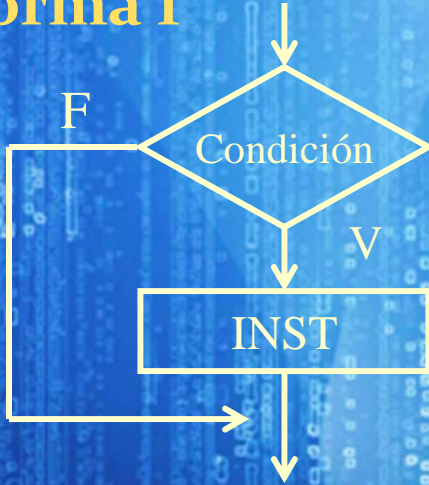
Estructuras Básicas de los Diagramas de Flujo

1) Secuencia Simple

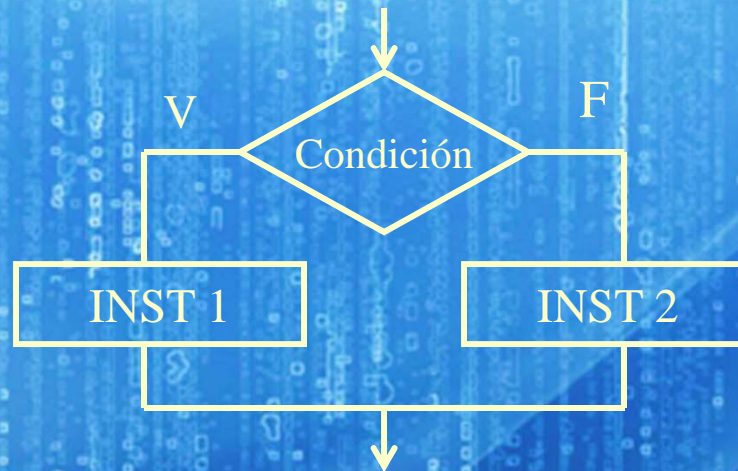


2) Decisión Simple

Forma 1

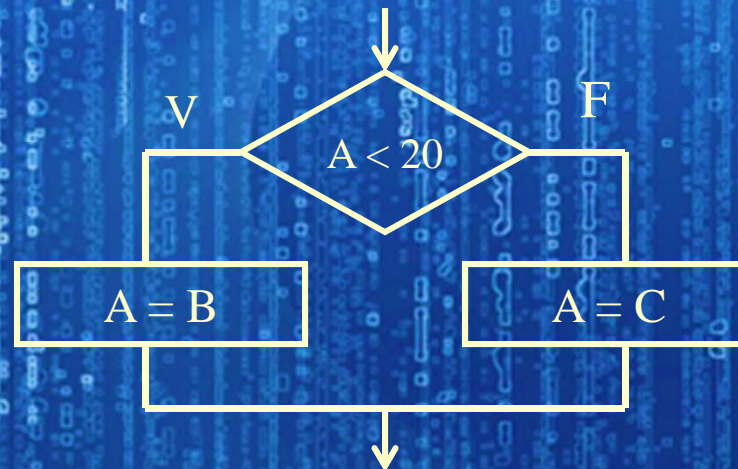
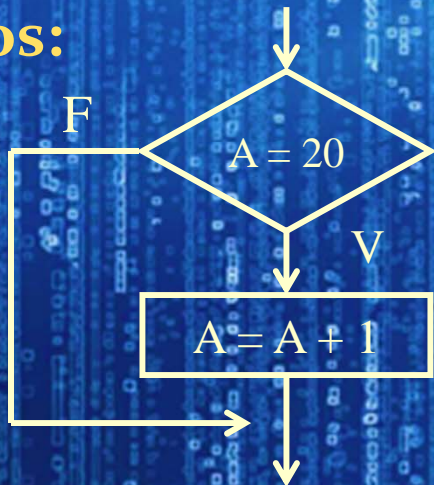


Forma 2

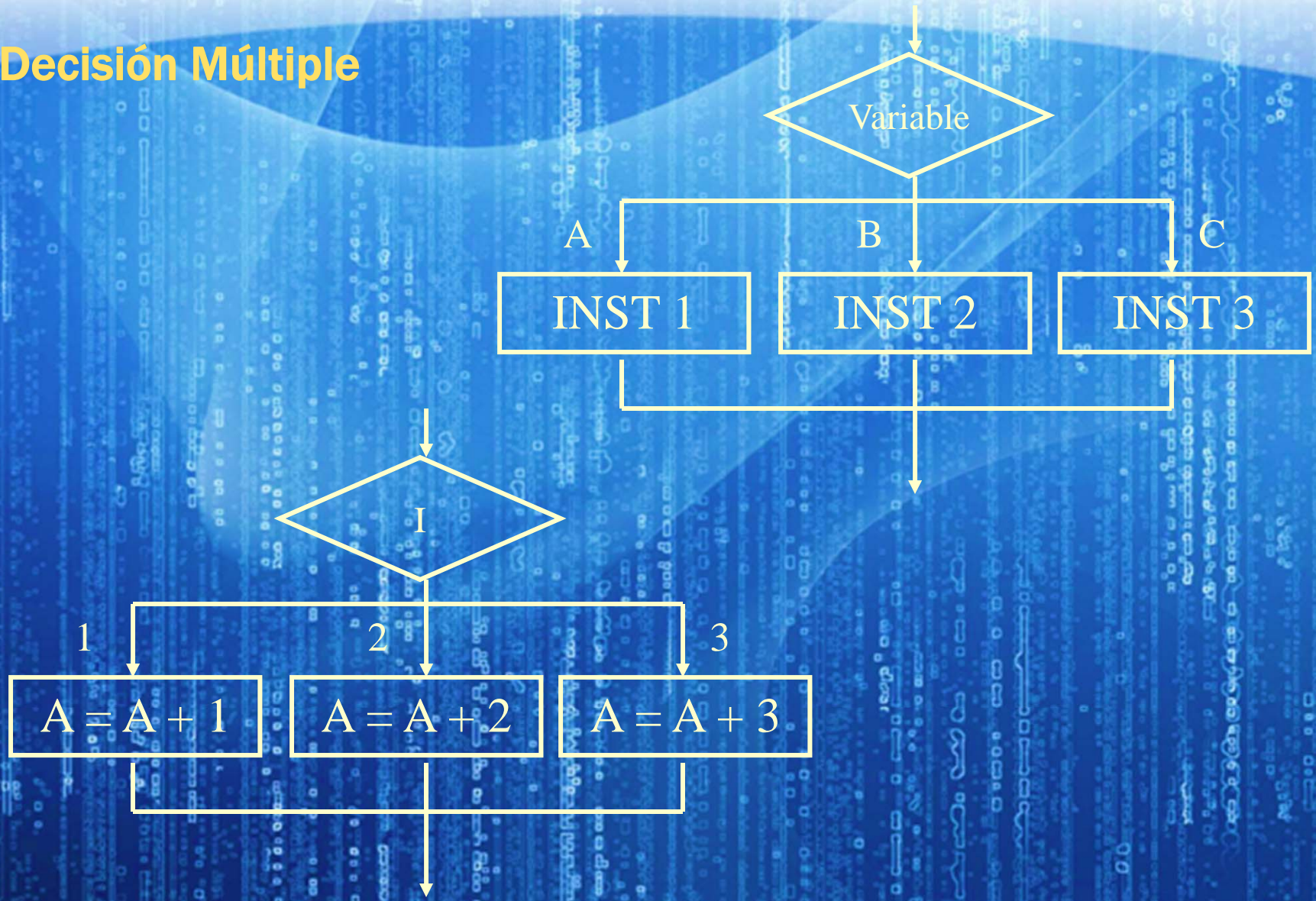


El objeto decisión selecciona el flujo a seguir de acuerdo al valor lógico de una condición. La condición debe ser siempre una expresión que al ser evaluada de como resultado un valor Lógico.

Ejemplos:



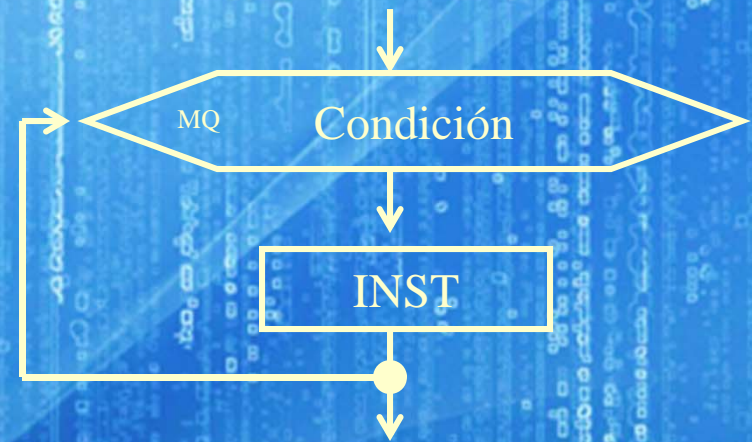
3) Decisión Múltiple



4) Iteración

El objeto **Ciclo Mientras** tiene como función el ejecutar un bloque de objetos mientras que una condición sea verdadera. La condición debe ser siempre una expresión que al ser evaluada de como resultado un valor de tipo Lógico.

Si al evaluar la condición se obtiene el valor F la ejecución del algoritmo continuará a partir del objeto que sigue al Cierre.

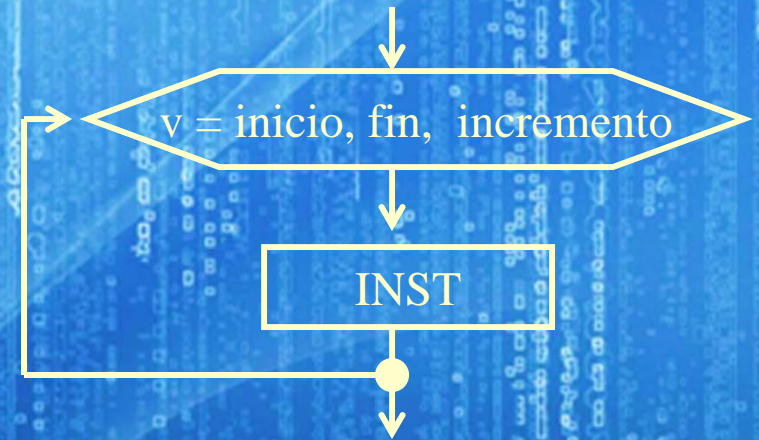


Ejemplos:

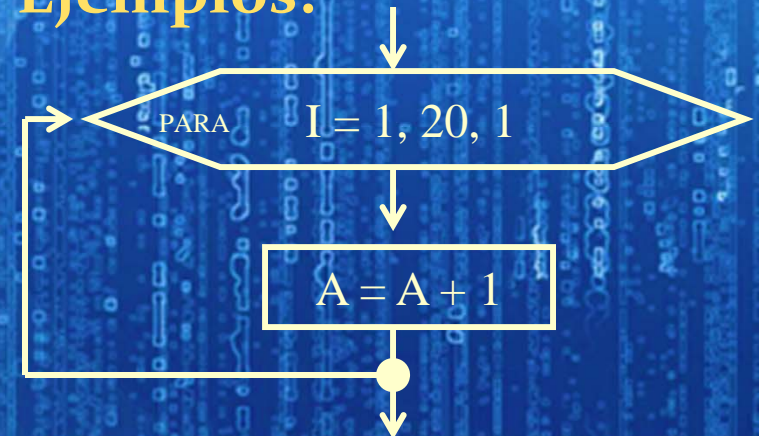


5) Repetición

Su función es ejecutar un bloque de objetos mientras que la variable contadora no alcance el límite establecido por el valor final. El contador es siempre una variable de tipo de dato entero. Contiene además un valor inicial que será asignado al contador al iniciar la ejecución del ciclo, un valor final y un valor de incremento. Si el contador excede el valor final, la ejecución continuará a partir del objeto que sigue al Cierre. En caso contrario, se ejecutará el cuerpo del ciclo y el contador será incrementado en el valor indicado por el incremento.

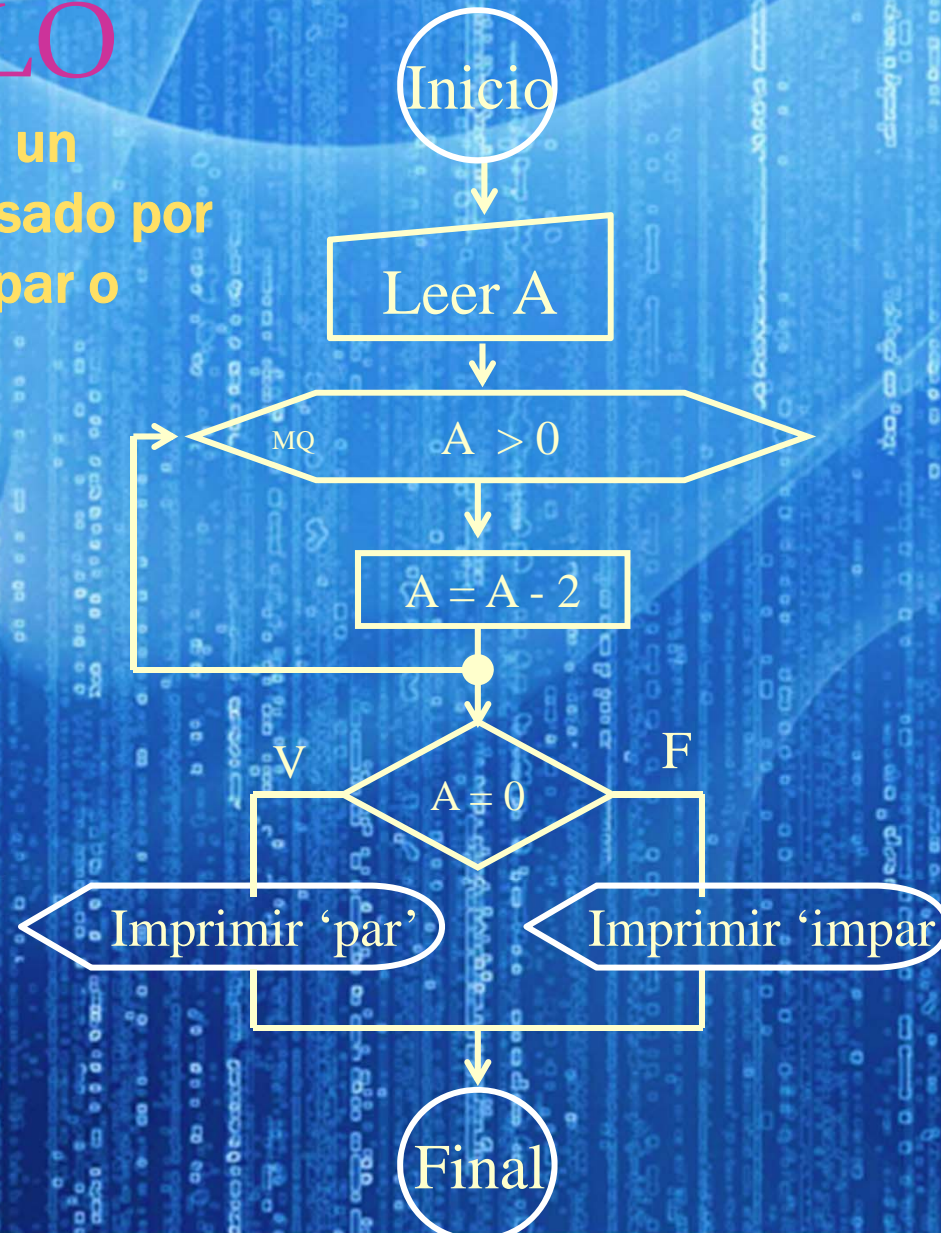


Ejemplos:



EJEMPLO

Determinar si un número ingresado por el usuario es par o impar.



Pseudocódigo

- Se denomina notación de pseudocódigo a aquella que permite describir la solución de un problema en forma de algoritmo dirigido a una computadora utilizando palabras y frases del lenguaje natural sujetas a determinadas reglas.
- El pseudocódigo se considera una herramienta para el diseño que permite obtener una solución mediante aproximaciones sucesivas.

Ejemplo 1

Suma el nro
5 y el nro 7
y muestra
por pantalla.

1. Inicio
2. `Int a, b, c // declara las variables a, b y c como enteras`
3. `a <- 5 // en la variable a se almacena el valor 5`
4. `b <- 7 // en la variable b se almacena el valor 7`
5. `c <- a + b // en la variable c se almacena la suma de a con b`
6. Escribir c
7. Fin

Algoritmo para calcular la suma de dos valores ingresados por teclado (Pseudocódigo)

1. Inicio
2. Int a, b, c
3. Escribir "Ingrese un valor entero"
4. Leer a
5. Escribir "Ingrese otro valor entero"
6. Leer b
7. $c \leftarrow a + b$
8. Escribir "La suma de"
9. Escribir a
10. Escribir "con"
11. Escribir b
12. Escribir "es"
13. Escribir c
14. Fin

Webgrafía y Licencia:

- Textos tomados, corregidos y modificados de diferentes páginas de Internet, tutoriales y documentos.
- Este documento se encuentra bajo Licencia Creative Commons 2.5 Argentina (BY-NC-SA), por la cual se permite su exhibición, distribución, copia y posibilita hacer obras derivadas a partir de la misma, siempre y cuando se cite la autoría del Prof. Matías E. García y sólo podrá distribuir la obra derivada resultante bajo una licencia idéntica a ésta.
- Autor:

Matías E. García

Prof. & Tec. en Informática Aplicada

www.profmatiasgarcia.com.ar

info@profmatiasgarcia.com.ar



creative
commons

