

# ALGORITMOS, DIAGRAMAS DE FLUJO Y PSEUDOCÓDIGO.

José Ortíz

17 de mayo de 2021



# CONTENIDO

- **Algoritmos.**
  - Características.
  - Partes fundamentales.
- **Diagramas de Flujo.**
  - Simbología.
- **Pseudocódigo.**
  - Estructura.
  - Consideraciones.
  - Ejemplos.
- **Conclusiones.**



# Algoritmos

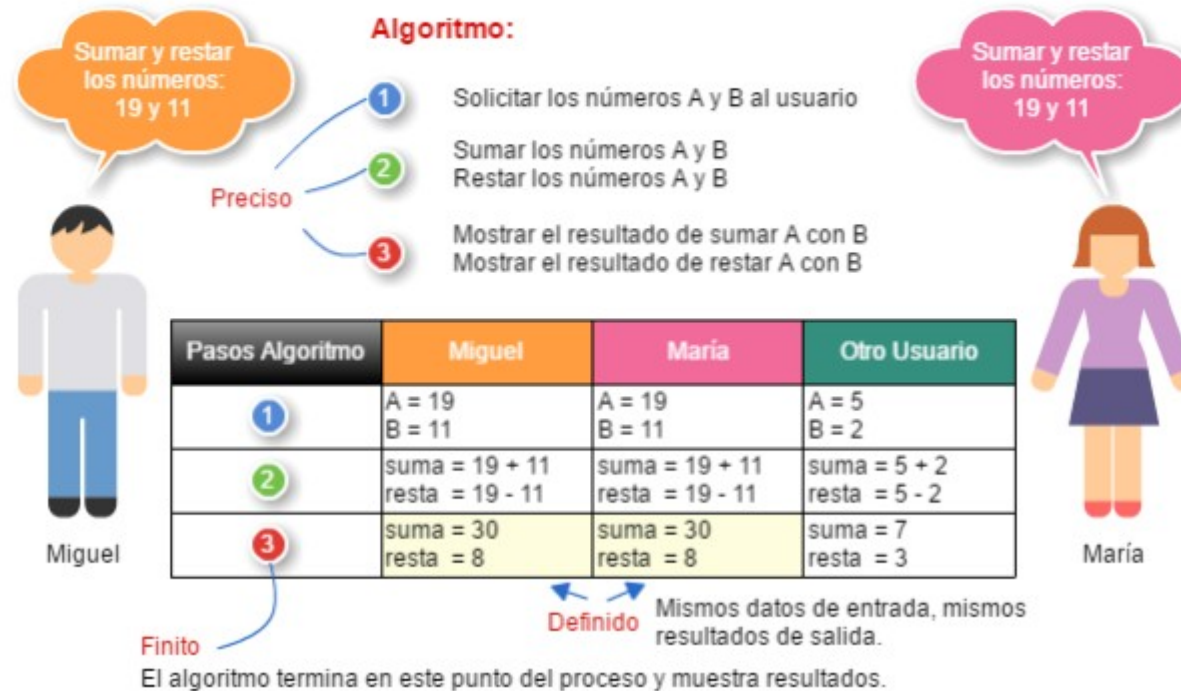
**Son las instrucciones o pasos finitos ordenados secuencialmente y libres de ambigüedad que debe llevar a cabo una computadora para lograr un resultado previsible.**

**Conjunto de pasos ordenados que permiten lograr un objetivo.**

# Algoritmos Características

- **Preciso**: Debe indicar el orden de ejecución de los pasos que conducen a la solución.
- **Definido**: El resultado nunca debe cambiar bajo las mismas condiciones.
- **Finito**: Existe un máximo de pasos a ejecutarse, el algoritmo debe terminar en algún momento.

# Algoritmos Características



# Algoritmos Partes Fundamentales

- **Entrada**: Datos necesarios para el inicio de la solución del problema.
- **Proceso**: Cálculos y operaciones necesarios para alcanzar la solución del problema.
- **Salida**: Información resultante la cual representa la solución del problema.

# Algoritmos Partes Fundamentales





- 1) Entrada: Solicitar los números A y B al usuario.**
- 2) Proceso: Sumar o Restar los números A y B.**
- 3) Salida: Resultado de la suma o resta de los números A y B.**

# Diagramas de Flujo

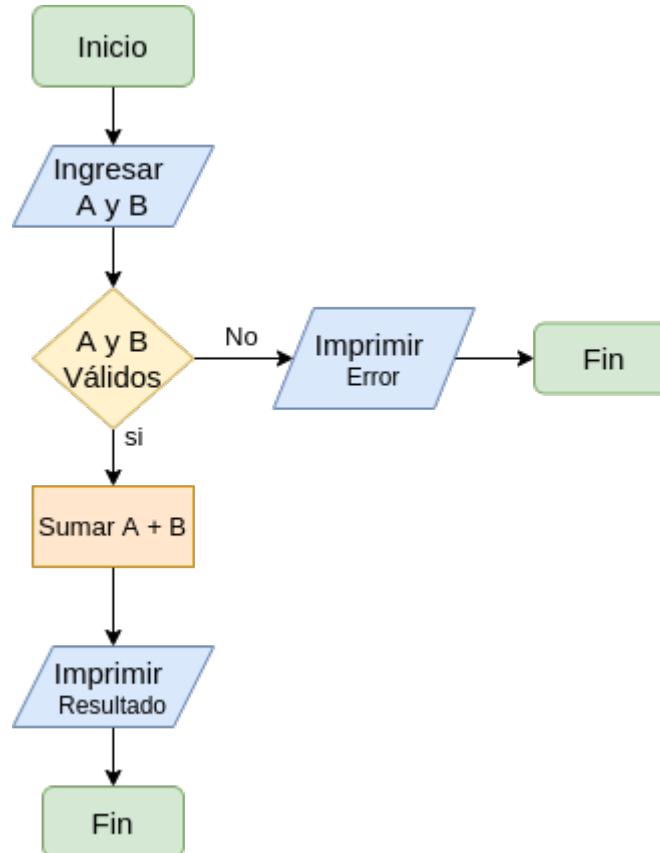
**Consiste en la representación grafica de un algoritmo, a través de una serie de pasos estructurados y vinculados que permite su revisión como un todo. Cada paso en el proceso es representado por un símbolo diferente.**



# Diagramas de Flujo Simbología

Símbolo	Nombre	Función
	Inicio/Final	Representa el inicio y final de un proceso.
	Linea de Flujo	Indica el orden de ejecución de las operaciones, la flecha indica la siguiente instrucción
	Entrada/Salida	Lectura de datos de entrada e impresión de datos en la salida.
	Proceso	Representa cualquier tipo de operación.
	Decisión	Permite analizar una situación con base a los valores Verdadero y Falso.

# Diagramas de Flujo Simbología



# Pseudocódigo

**Es un lenguaje de especificación de algoritmos, utiliza una notación(Palabras claves reservadas) similar a la de un lenguaje de programación estándar.**

**Las palabras claves que utiliza están escritas en lenguaje castellano(Lenguaje natural estandarizado salvo las formulas matematicas). Estas especifican las acciones a ejecutar para resolver un problema.**

# Pseudocódigo Estructura

- **Declaraciones:** Apartado donde se definen los insumos de entrada ya sea en forma de Variable o Constante.
- **Palabras Reservadas:** Inicio, Leer, Escribir, si, sino, mientras, para, repetir.
- **Condicionales:** Estructuras de control encargadas en evaluar una condición y ejecutar una decisión.
- **Ciclos:** Estructuras de control dada una condicion ejecutar cierta instrucción.

# Pseudocódigo Consideraciones

- **La declaración del pseudocódigo es de arriba hacia abajo y de izquierda a derecha.**
- **Debe tener una sentencia de inicio y fin.**
- **Todo ciclo o condicional debe tener un cierre.**
- **Toda sentencia termina en (;).**
- **La declaración de Constantes no permite su cambio de valor bajo ninguna circunstancia.**

# Pseudocódigo Ejemplo



```
1  Inicio
2      var a;
3      var b;
4      var resultado;
5      leer a;
6      leer b;
7      si a == "es un numero" Y b=="es un numero" entonces
8          resultado = a + b;
9      fsi
10     escribir resultado;
11  fin
```

# Pseudocódigo Ejemplo



```
1  Inicio
2      var entrada;
3      si entrada == "no es un numero" entonces
4          escribir "Error entrada debe ser un numero";
5      fsi
6      si entrada == "es un numero" entonces
7          var inicio = 0;
8          mientras (entrada > inicio ) hacer
9              escribir "El valor de entrada es= " + entrada;
10             inicio = inicio + 1;
11          fmientras
12      fsi
13  fin
```

# Conclusiones

- Los algoritmos deben ser bien definidos y acotados su alcance.
- Los diagramas de flujo son útiles, sin embargo si el algoritmo es muy complejo puede ocasionar confusiones.
- Se debe tener cuidado de respetar los estándares del pseudocódigo, si bien no es un lenguaje de programación estos ayudan a definir la lógica antes de ser traducidos a un lenguaje de programación