



Tecnólogo en Análisis y Desarrollo de Software

Sesión en Línea



www.sena.edu.co

Instructor Julián Giraldo



Fundamentos de Programación Estructurada

Análisis y Solución de Problemas Aplicando Algoritmos

Instructor Julián Giraldo

Temáticas



- 1. Introducción a la Algoritmia – Conceptualización**
- 2. Métodos para la Resolución de Problemas**
- 3. Herramientas para la representación de Algoritmos**
- 4. Estructuras Algorítmicas de Control**
- 5. Programa PSeInt – IDE**
- 6. Ejemplos Base Algoritmos**
- 7. Actividad a Realizar y Entregables (Individual)**
- 8. Planteamiento de Dudas o Inquietudes**

Algoritmo: “Conjunto ordenado y finito de operaciones que permite hallar la solución de un problema”.

Formalmente definimos un **Algoritmo** como: “un conjunto de pasos, procedimientos o acciones que nos permiten alcanzar un resultado o resolver un problema”. Ejemplos: Receta de cocina, Cambio de una llanta de un carro, Utilizar un cajero electrónico.

Problema: Es el desconocimiento de los pasos o acciones a seguir para solucionar una determinada situación, sus componentes son: las metas, los datos, las restricciones y los métodos.

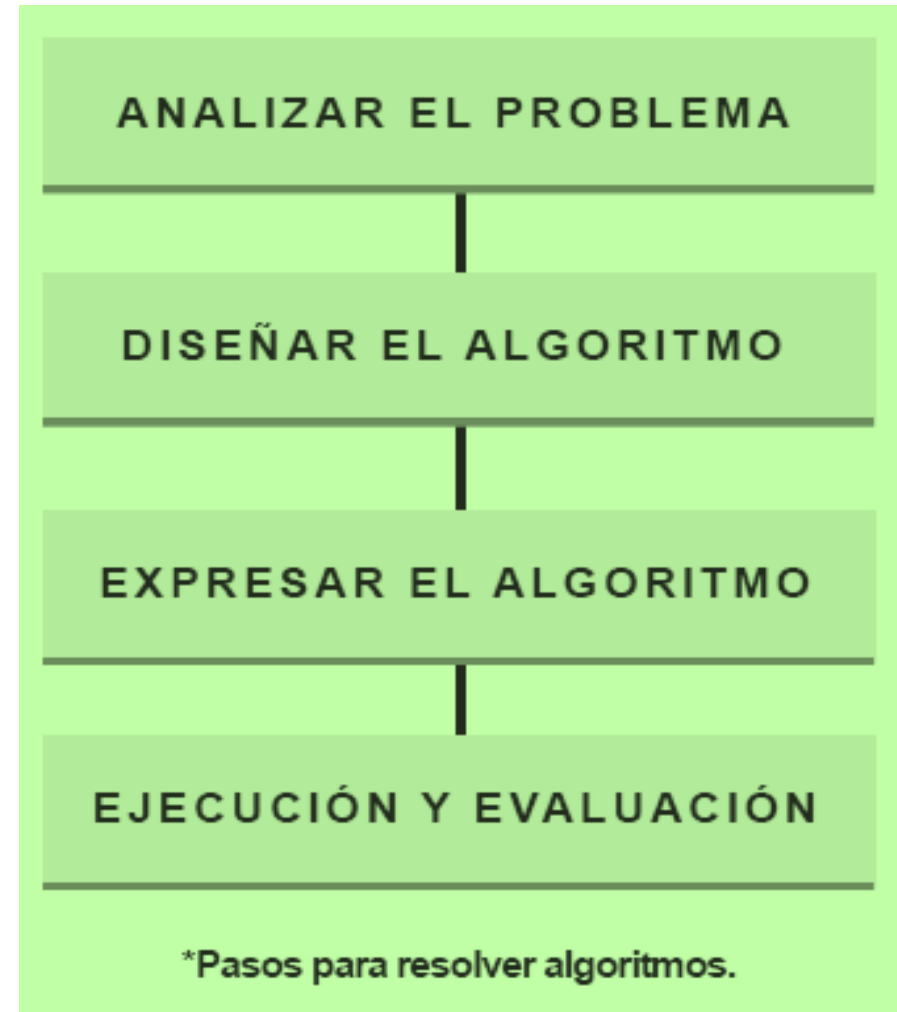
Métodos para la Resolución de Problemas



Método: “Conjunto de pasos o acciones para alcanzar un resultado específico”.

Existen **tres métodos** para la resolución de problemas:

- Particularizar y Generalizar
- **Divide y Vencerás**
- Ensayo y Error

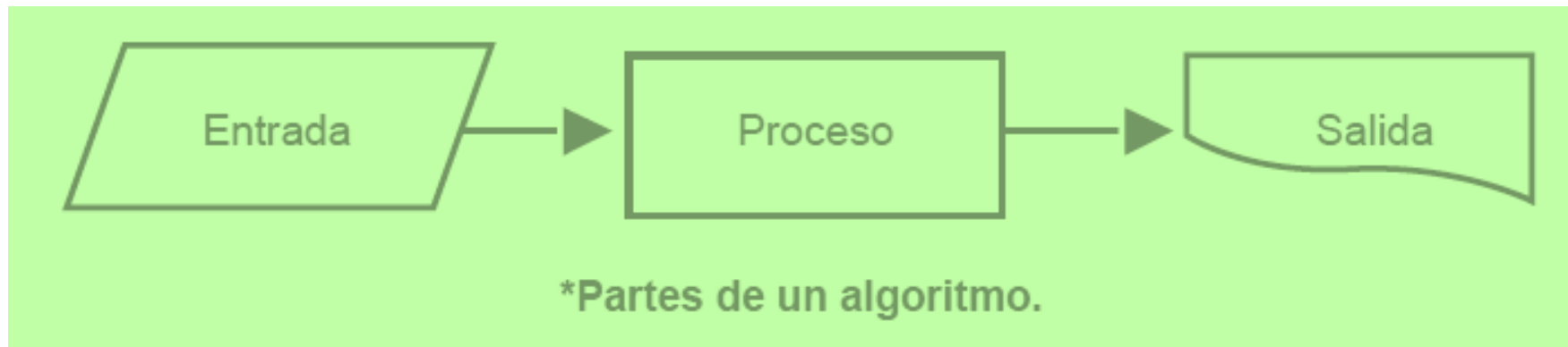


Características de los Algoritmos



Un algoritmo debe ser:

- Preciso e indicar el orden de realización de cada paso.
- Estar definido. Si se sigue un algoritmo dos veces o más, se debe obtener el mismo resultado cada vez.
- Ser finito. Si se sigue un algoritmo, se debe terminar en algún momento.
- Debo contener cómo mínimo las siguientes partes:



Herramientas para la Representación de Algoritmos



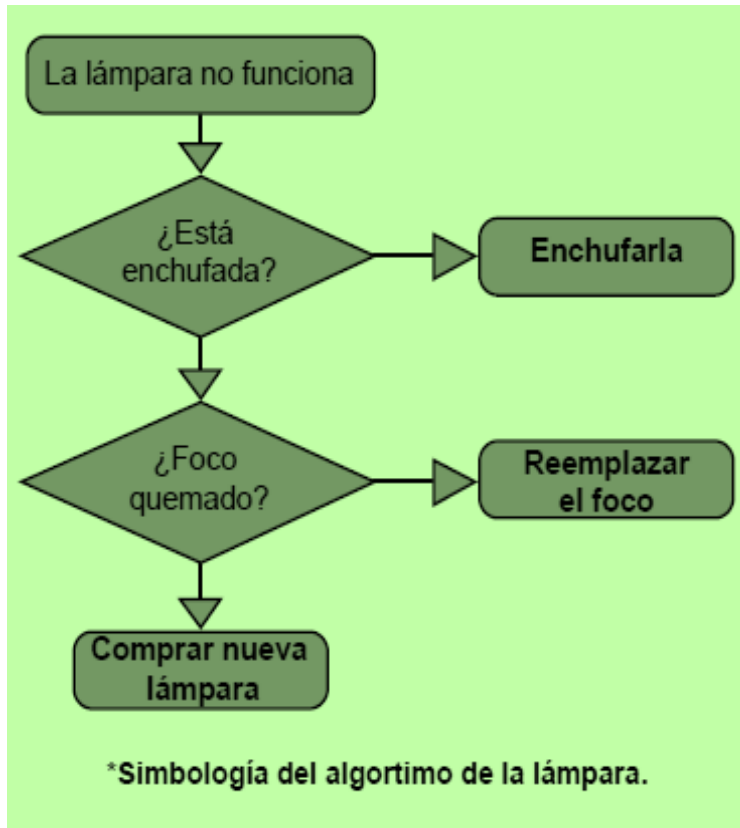
Existen diferentes tipos de herramientas para la representación de algoritmos: Lenguaje Natural, Diagramas de Flujo y **Pseudocódigo**.

Algoritmo: cargar un celular.

INICIO

1. Buscar el cargador.
2. Agarrar el cargador.
3. Agarrar el celular.
4. Poner el enchufe en el celular.
5. Conectarlo al tomacorriente.
6. Esperar a que se ponga a cargar.
7. Desconectarlo del tomacorriente.
8. Ponerlo en un lugar seguro.

FIN



ALGORITMO área de un triángulo;

VAR

ENTERO divisor;

REAL area;

REAL base;

REAL altura;

INICIO

divisor = 2;

ESCRIBIR("Escriba la base ");

LEER(base)

ESCRIBIR("Escriba la altura ");

LEER(altura);

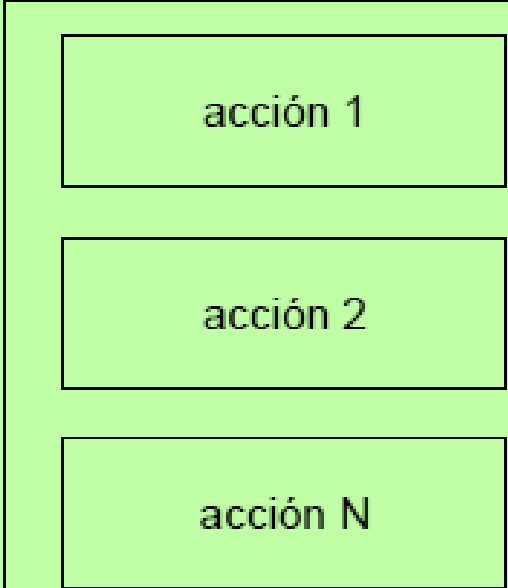
area = base * altura / divisor;

ESCRIBIR(area);

FIN

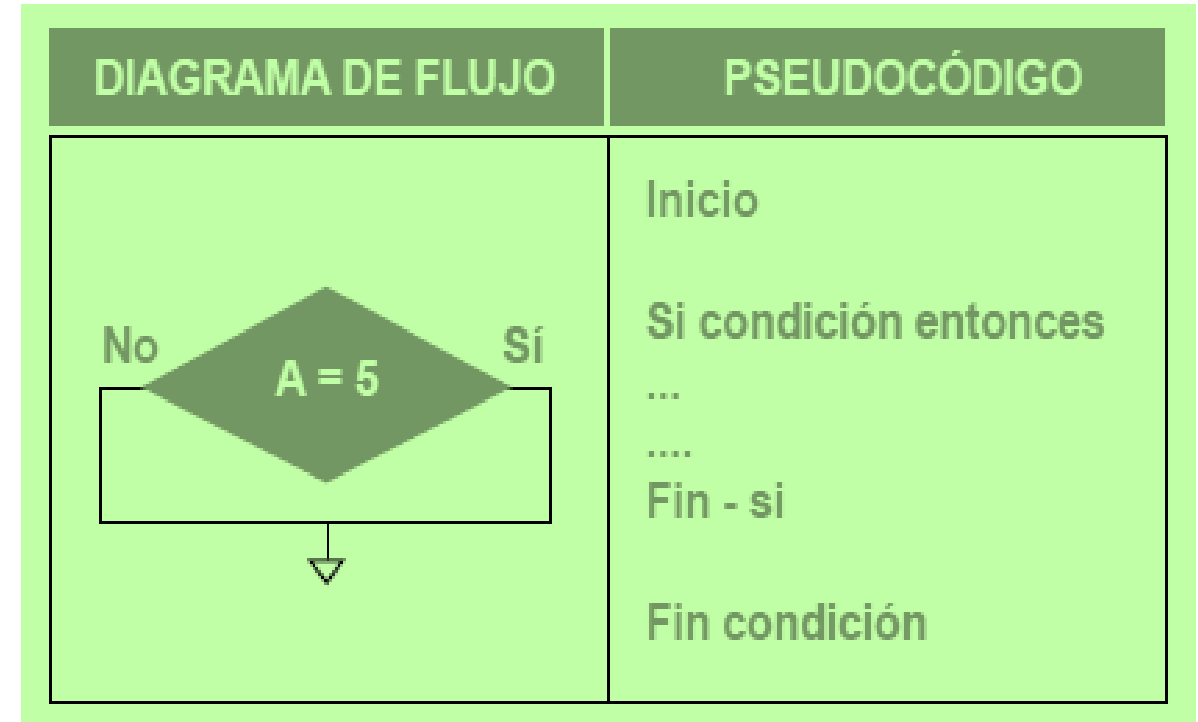
Estructura Secuencial o Lineal:

Es una de las estructuras más sencillas, conocida también como estructura lineal y se compone de instrucciones que deben ejecutarse secuencialmente.

DIAGRAMA DE FLUJO	PSEUDOCÓDIGO
 <pre>graph TD; A[acción 1] --> B[acción 2]; B --> C[acción N];</pre>	<pre>Inicio <acción1> <acción2> * * * <acciónN> Fin</pre>

Estructura Condicional o Decisión:

Son utilizadas para tomar decisiones lógicas, llamadas también estructuras selectivas o alternativas, en ellas se evalúa una condición y en función del resultado de la misma se realiza una opción u otra.



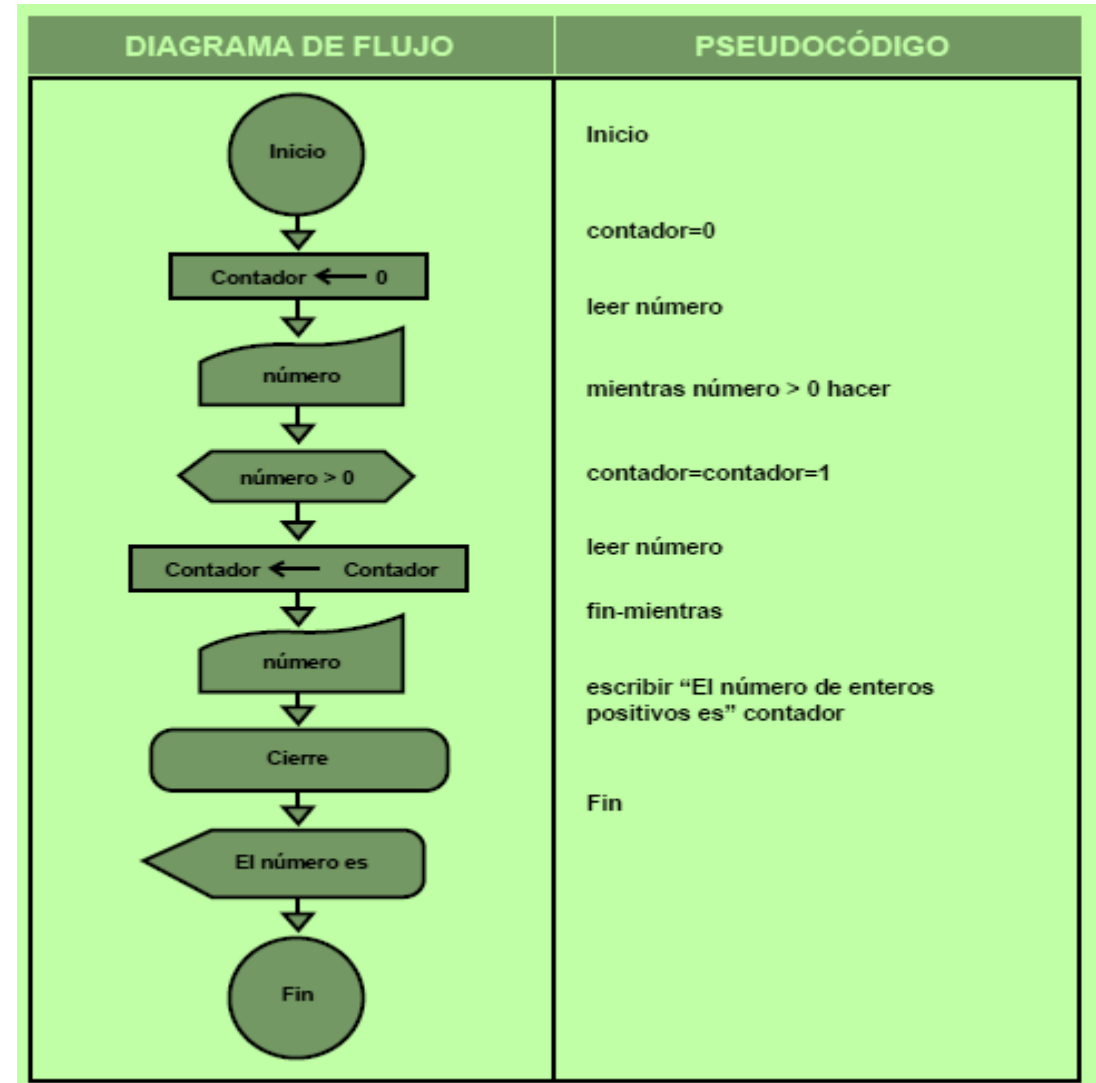
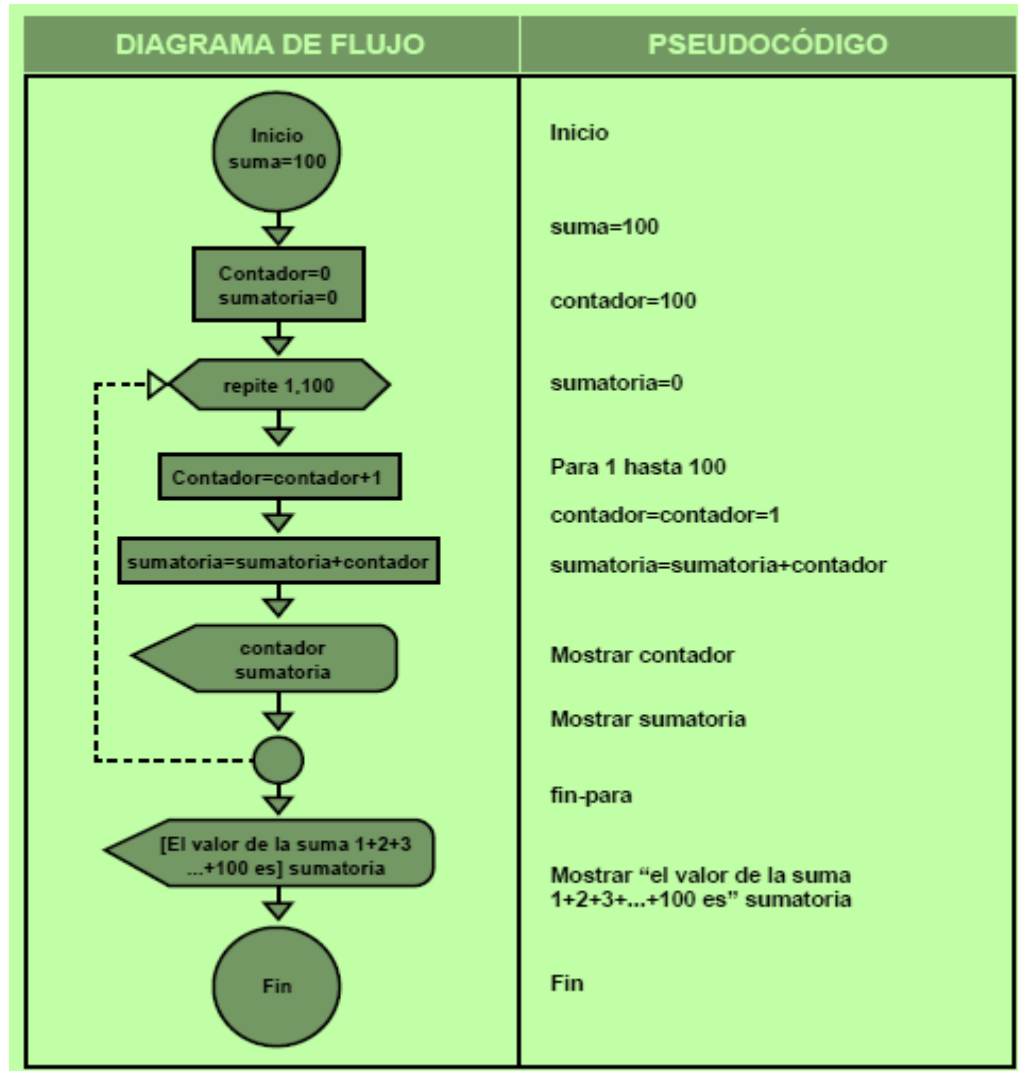
Estructuras Cíclicas y/o Repetitivas (Bucles)

Un tipo muy importante de estructura es el algoritmo necesario para **repetir una o varias acciones un número determinado de veces**. Las estructuras que repiten una secuencia de instrucciones se denominan bucles, y se llama iteración al hecho de repetir la ejecución de una secuencia de acciones.

En otras palabras, el algoritmo repite muchas veces las acciones. Las dos principales preguntas a realizarse en el diseño de un bucle son:

- ¿Qué instrucciones contiene el bucle?
- ¿Cuántas veces se deben repetir?.

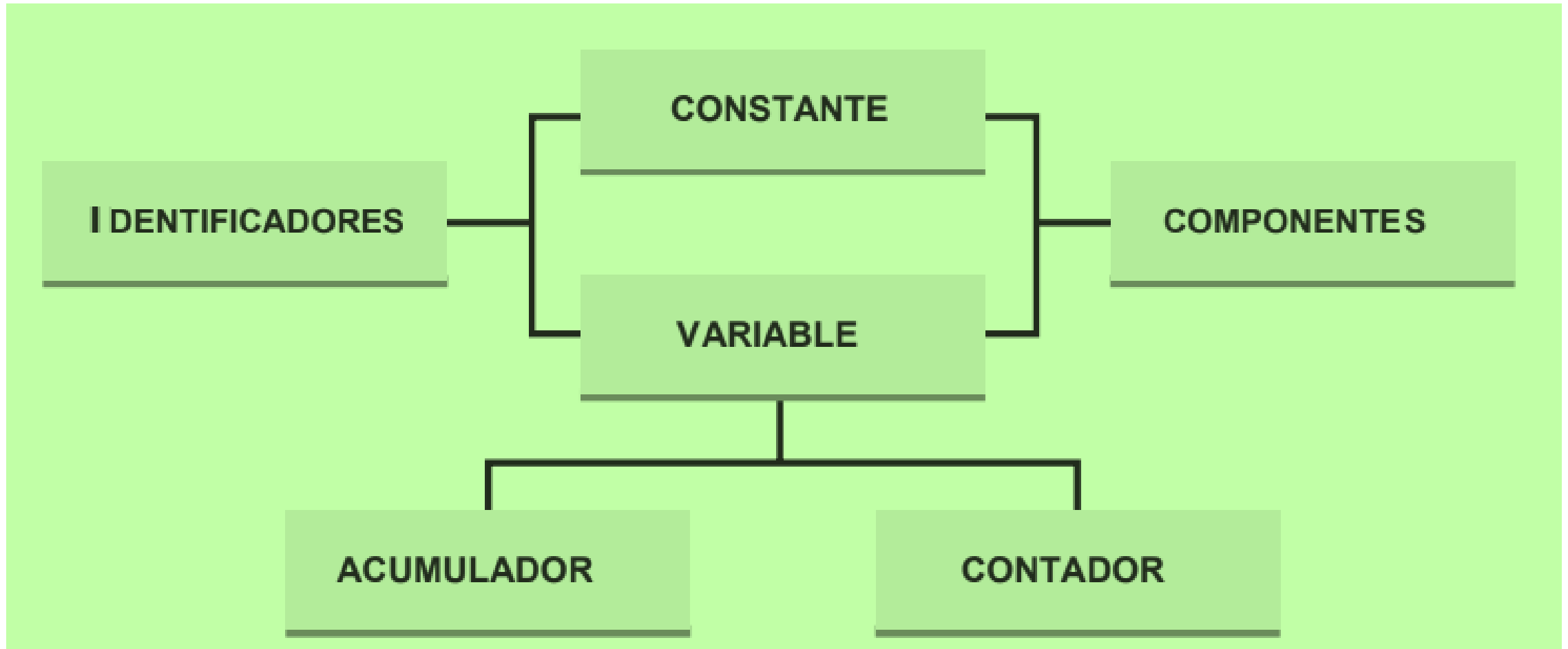
Estructura Algorítmicas de Control (Secuencial–Condicional–Iteración)



PSeInt es una herramienta educativa que ayuda a aprender a programar. Utiliza un lenguaje de programación **basado en pseudocódigo**, que es una descripción informal de un algoritmo.

Características de PSeInt:

- Permite escribir algoritmos en pseudocódigo
- Genera y edita diagramas de flujo
- Interpreta (ejecuta) los algoritmos escritos
- Permite modificar el algoritmo
- Permite ver los cambios en la ejecución inmediatamente
- Incluye herramientas de edición como autocompletado, plantillas de comandos, y resaltado de bloques lógicos
- Ofrece diferentes perfiles de configuración predefinidos



Los operadores: son símbolos que sirven para manipular datos. Y en un algoritmo es necesario dejar un espacio en blanco a cada lado del signo aritmético. Los operadores y las funciones que realizan se clasifican como se indica a continuación:

- **Aritméticos:** permiten realizar operaciones entre datos de tipo numérico y dan como resultado otro valor de tipo numérico. Ejemplo: producto o multiplicación (*); división (/); suma (+); resta (-); asignación (=).

- **Alfanuméricos:** permiten operar con datos de tipo carácter (letras) o cadenas de texto. La mayoría de los lenguajes de programación admiten el operador + para realizar la concatenación (unión) de caracteres o cadenas.

- **Relacionales:** permiten la comparación entre datos del mismo tipo de dato y dan como resultado dos valores posibles: verdadero o falso. Ejemplo: igual a (=); menor que (<); mayor que (>,) mayor e igual que (>=).

- **Lógicos:** posibilitan la evaluación lógica de dos expresiones de tipo lógico. Dan como resultado uno de dos valores posibles: verdadero o falso. Ejemplo: negación (no); conjunción (y); disyunción(o).

Preguntas





GRACIAS

Línea de atención al ciudadano: 01 8000 910270
Línea de atención al empresario: 01 8000 910682



@SENAComunica

www.sena.edu.co