	<b>INSTITUCIÓN EDUCATIVA JOSÉ EUSEBIO CARO</b>  <b>Tecnología e informática</b>
<b>Docente</b>	Jesús Eduardo Madroñero Ruales
<b>Propósito del taller</b>	Comprender los métodos de representación, instrucciones, reglas y diagramas básicos para la representación de algoritmos. Aplicar los métodos de representación de algoritmos, en la solución de problemas cotidianos.
<b>Competencias</b>	Análisis del funcionamiento de prototipos conformados por artefactos y procesos como respuesta a necesidades o problemas.

### Representación de algoritmos: Diagrama Rectangular y Pseudocódigo

Los algoritmos deben ser representados usando algún método que les permita ser independizados del lenguaje de programación que se requiera utilizar. Los métodos más usuales son: diagramas de flujo (tema anterior), diagramas rectangulares y pseudocódigos.

**El diagrama rectangular:** No utiliza flechas de conexión, sino que se construye, fundamentalmente, con tres figuras que representan las estructuras de control de la programación estructurada.

<b>Inicio</b>
<b>Escriba:</b> "digite año inicial"
<b>Lea:</b> inicio
<b>Escriba:</b> "digite año final"
<b>Lea:</b> final
<b>resultado</b> = final – inicio
<b>meses</b> = resultado * 12
<b>Escriba:</b> "el número de meses que han pasado es:", <b>meses</b>
<b>Fin</b>

Un diagrama rectangular empieza con un rectángulo vacío que se va llenando de arriba hacia abajo, en la medida en que es necesario representar un determinado paso. No tiene figuras especiales para demarcar los medios de entrada y de salida, pero dentro de la figura escogida para representar la instrucción se escribe la acción a ejecutar. Los bloques de inicio y fin del algoritmo pueden omitirse ya que se asume que el orden de ejecución es siempre de arriba hacia abajo, sin embargo, se recomienda no hacerlo.

**El pseudocódigo:** El pseudocódigo es la representación de los pasos del algoritmo a través de palabras, utilizando una nomenclatura estandarizada para denotar el significado de cada paso.

Dentro de éste es permitido la sangría con el fin de que se visualice, en forma sencilla, el grupo de instrucciones pertenecientes a una determinada acción. Esta herramienta resulta muy útil para el seguimiento de la lógica de un algoritmo y, sobre todo, facilita la transcripción a un lenguaje de programación. Tiene la desventaja de que el programador trata de escribir los pasos del algoritmo utilizando palabras reservadas. Es necesario, entonces, al igual que en el diagrama de flujo, establecer unos parámetros o formas de expresar cada instrucción.

#### **Inicio**

**Escriba:** "digite año inicial"

**Lea:** inicio

**Escriba:** "digite año final"

**Lea:** final

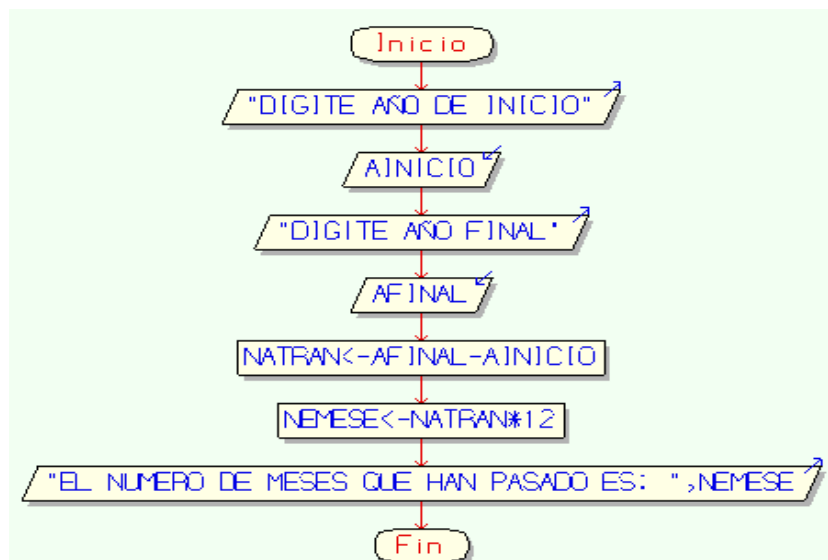
**resultado** = final – inicio

**meses** = resultado \* 12

**Escriba:** "el número de meses que han pasado es:", **meses**

#### **Fin**

Nótese cómo el grupo de instrucciones entre las palabras inicio y fin se han rodado un poco hacia la derecha (sangría), lo cual permite visualizar en una forma sencilla el grupo de instrucciones pertenecientes a ese bloque. El otro esquema ya visto anteriormente (ver guía pasada), es el diagrama de flujo. En la siguiente figura se detalla el diagrama de flujo para el ejemplo:



**Figura 1 - Ejemplo de Diagrama de Flujo**

### Consejos a tener en cuenta:

- Trate de entender y hacer el mejor análisis posible del problema a solucionar.
- A todos los campos utilizados en un programa se les asigna una única dirección de memoria.
- Los campos variables (variables) se les asignan valores en el momento de ejecución, bien sea a través de una instrucción de entrada de datos o una instrucción de asignación.
- Cuando se le asigna una variable; lo que había antes en esa determinada dirección es destruido, no se puede recuperar.
- No todos los valores de partida al hacer el algoritmo son datos de entrada, sólo lo son aquellos valores que desconocemos en el momento, pero que serán suministrados en la fase de ejecución.
- Al extraer de memoria uno o más valores hacia un medio externo, éstos permanecen; no se borran de la memoria de la computadora.
- Usted aprende a construir algoritmos haciendo algoritmos. Trate de hacer el máximo número de algoritmos propuestos. Busque, si es necesario, enunciados en la red o en otros textos.

### Recursos complementarios

- [1]** DiscoDurodeRoer (2015, 27 de abril). Ejercicios PseInt – Básicos #1 – Empezando por lo básico [video]. <https://youtu.be/DHli4dcaMEc>
- [2]** JuanRa Garcia Montes (2020, 26 de marzo). Programación: Diagramas de flujo y pseudocódigo [video]. <https://youtu.be/Lub5qOmY4JQ>
- [3]** Eliezer De León (2016, 21 de septiembre). [PSEINT] Diagrama de flujo y Pseudocódigo que determina tu edad [video]. <https://youtu.be/ocUWlylurUo>
- [4]** GabyB (2018, 23 de mayo). Tutorial – Como hacer una Diagrama de flujo en PseInt [video]. <https://youtu.be/RTWzhECFNyc>

### Actividad conceptual

**Nota:** Los siguientes algoritmos, deben ser resueltos de las tres formas de representación detalladas: Diagrama de flujo, diagrama rectangular y pseudocódigo.

1. Realizar un cuadro comparativo entre las tres formas de representación de algoritmos: los diagramas de flujo, los diagramas rectangulares y los pseudocódigos.
2. ¿Cuál de las tres herramientas para elaborar algoritmos le parece más eficaz? ¿Por qué?
3. Elaborar un algoritmo que convierta un dato de entrada, de centímetros a pulgadas. Recuerde que una pulgada es igual a 2,54 centímetros. ¿Es posible resolver este problema en Excel? Argumentar su respuesta.
4. Elaborar un algoritmo que permita convertir grados centígrados en:
  - a. Grados Fahrenheit.
  - b. Grados Kelvin.
5. Elaborar un algoritmo que calcule el promedio de cinco notas ingresadas por un usuario. ¿Es posible resolver este problema en Excel? Argumentar su respuesta.