

RESOLUCIÓN DE PROBLEMAS
COMPUTACIONALES III
HERRAMIENTAS PARA LA REPRESENTACIÓN
DE ALGORITMOS

FLUJO DE CONTROL DE UN ALGORITMO

- Es el orden temporal en el cual se ejecutan los pasos individuales del algoritmo.
- El flujo normal de un algoritmo es el flujo lineal o secuencial de los pasos (un paso a continuación de otro).
- Para apartarse del flujo normal lineal están las estructuras de control de repetición y selección:
 - Una permite repetir automáticamente un grupo de pasos (repetición).
 - Otra permite seleccionar una acción de entre un par de alternativas específicas, teniendo en cuenta determinadas condiciones (selección).

Teorema de Böhm y Jacopini

- También llamado “Teorema Fundamental de la Programación Estructurada”.
- El teorema de Böhm - Jacopini [C.Böhm, G.Jacopini, Comm. ACM vol.9, nº5, 366-371, 1966] establece que:
- **“Todo programa propio se puede escribir utilizando únicamente las estructuras de control secuencial, condicional e iterativa”.**
- Un programa se define como propio si cumple las siguientes condiciones:
 - Si tiene un solo punto de entrada y un solo punto de salida
 - Existen caminos desde la entrada hasta la salida que pasan por todas las partes del programa.
 - Todas las instrucciones son ejecutables y no existen bucles sin fin.

HERRAMIENTAS PARA LA REPRESENTACIÓN DE ALGORITMOS

- Diagramas de Bloques
- Diagramas de Flujo.
- Pseudocódigo.
- Lenguaje natural
- Tablas de Decisión

DIAGRAMAS DE BLOQUES

- La estructura de un algoritmo se puede representar con un diagrama estructurado en forma de bloques, donde se muestren los sucesivos refinamientos a partir del problema inicial.

DIAGRAMAS DE BLOQUES

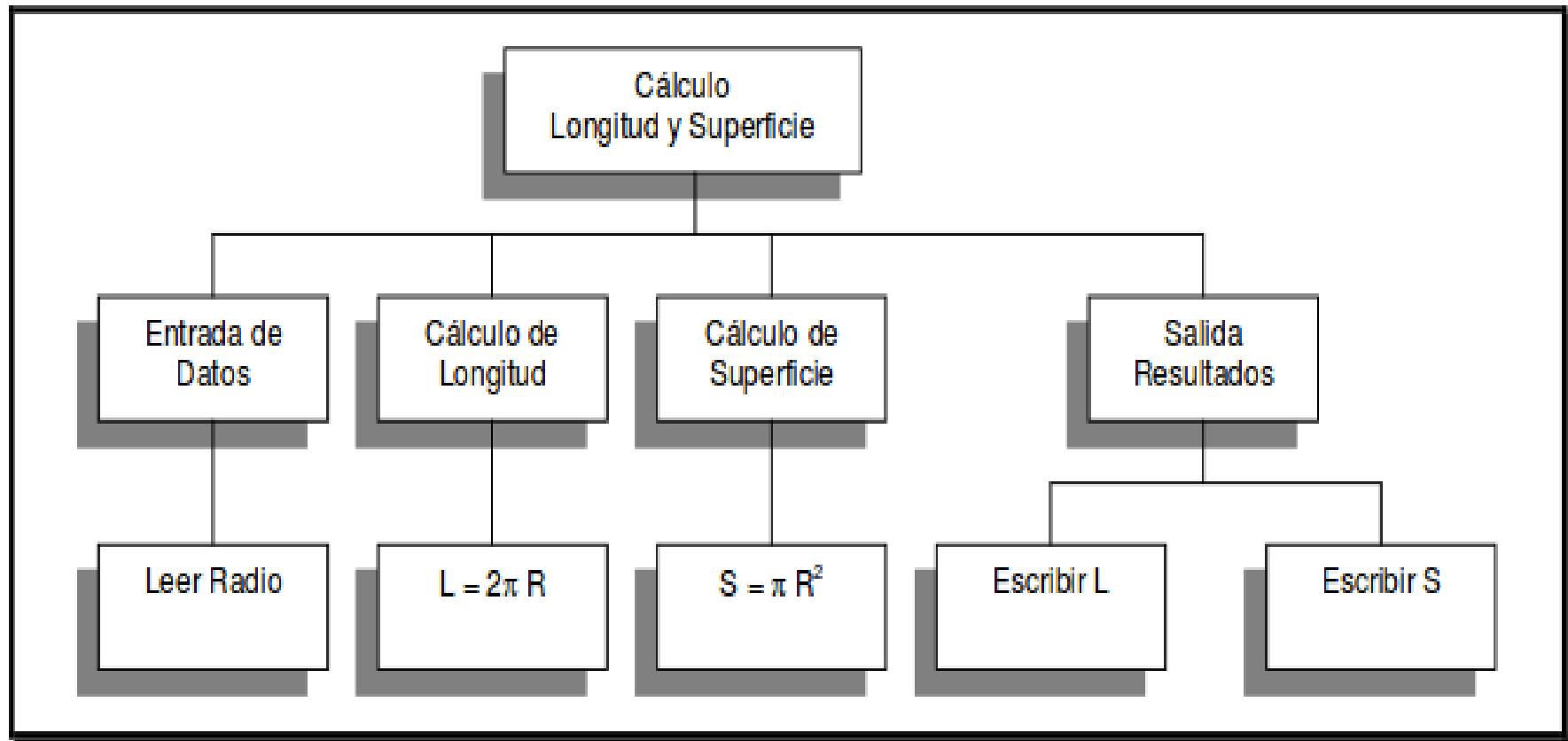


Figura 3. Diagrama de bloques para el cálculo de la superficie y longitud del círculo

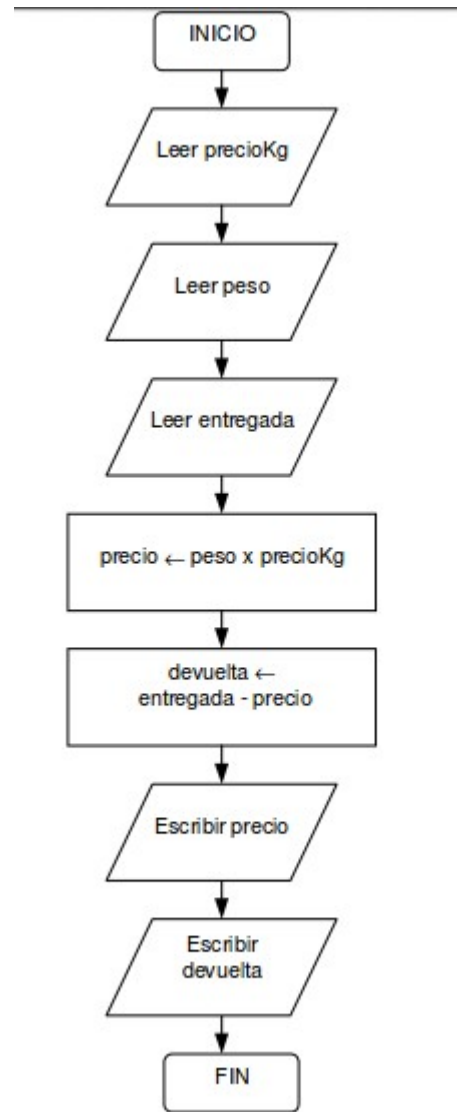
DIAGRAMAS DE FLUJO

- Los diagramas de flujo son una herramienta gráfica para descripción de algoritmos.
- Un diagrama de flujo muestra la lógica del algoritmo.
- Los diagramas de flujo han sido una de las herramientas de programación clásica más usadas.
- Ventajas
 - Muestran el flujo lógico del algoritmo de una manera clara
- Inconvenientes
 - Su complejidad aumenta con la complejidad del problema
 - Son difíciles de actualizar y oscurecen la estructura del algoritmo.

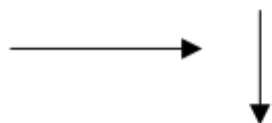
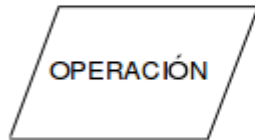
DIAGRAMAS DE FLUJO

- Un diagrama de flujo debe reflejar:
 - El comienzo del programa
 - Las operaciones que el programa realiza
 - El orden en que se realizan
 - El final del programa

DIAGRAMAS DE FLUJO



DIAGRAMAS DE FLUJO



Indican el inicio y el fin del algoritmo. Del primero sólo puede salir una línea y al segundo sólo puede llegar una línea.

Indica una acción a realizar. Tiene una única línea de entrada y otra de salida.

Indica una acción de entrada (lectura de datos desde el teclado) o salida (escritura de un dato por pantalla). Tiene una única línea de entrada y otra de salida.

Se usan para tomar una decisión, dependiendo de cierta condición. Tiene una línea de entrada y dos de salidas (una por la cual salir, si la condición es cierta, y otra, si es falsa).

Se usan para conectar los símbolos anteriores y especifican el flujo del algoritmo.

Se usan para conectar distintos puntos de los diagramas de flujo, evitando cruces de flechas.

DIAGRAMAS DE FLUJO

- En la representación es conveniente seguir las siguientes reglas:
 - El comienzo del programa figurará en la parte superior del diagrama.
 - Los símbolos de comienzo y fin deberán aparecer una única vez.
 - El flujo de las operaciones será de arriba a abajo y de izquierda a derecha.
 - Se debe guardar cierta simetría en la representación de bifurcaciones y bucles.
 - Se evitarán los cruces de líneas de flujo, utilizando conectores.

PSEUDOCÓDIGO

- Es un lenguaje algorítmico que permite representar las construcciones básicas de los lenguajes de programación pero manteniéndose próximo al lenguaje natural.
- Ventajas
 - Podemos centrarnos sobre la lógica del problema olvidándonos de la sintaxis de un lenguaje concreto.
 - Es fácil modificar el algoritmo descrito.
 - Es fácil traducir directamente a cualquier lenguaje de programación el algoritmo obtenido.

PSEUDOCÓDIGO

INICIO

LEER precioKg

LEER peso

LEER pago

$\text{precio} \leftarrow \text{peso} \times \text{precioKg}$

$\text{devuelta} \leftarrow \text{pago} - \text{precio}$

ESCRIBIR precio

ESCRIBIR devuelta

FIN

PSEUDOCÓDIGO

- El pseudocódigo que nosotros utilizaremos consta exactamente de las siguientes construcciones:
 - INICIO : Indica el comienzo del algoritmo
 - FIN : Indica la finalización del algoritmo.
 - LEER : Se usa para leer un dato del teclado.
 - ESCRIBIR : Se usa para escribir un dato por pantalla.

PSEUDOCÓDIGO

- SELECCIÓN -
CONDICIÓN
- REPETITIVA -
ITERATIVA

```
SI <condición> ENTONCES  
    <Acción 1>  
EN OTRO CASO  
    <Acción 2>  
FINSI
```

```
MIENTRAS <Condición> HACER  
    <Acción 1>  
FINMIENTRAS
```

TABLAS DE DECISIÓN

- Presenta:
 - Las posibles situaciones o condiciones que pueden plantearse en un problema,
 - Las respectivas acciones o tratamiento a seguir en cada caso

CONDICIONES	1	2	3	4
Antigüedad empleado	<5 años	5 a <10 años	10 a 15 años	> 15 años
ACCIONES				
Calcular bonificación por antigüedad.				
Sueldo x 1% x años antig.	X			
Sueldo x 1,5% x años antig.		X		
Sueldo x 2% x años antig.			X	
Sueldo x 2,5% x años antig.				X