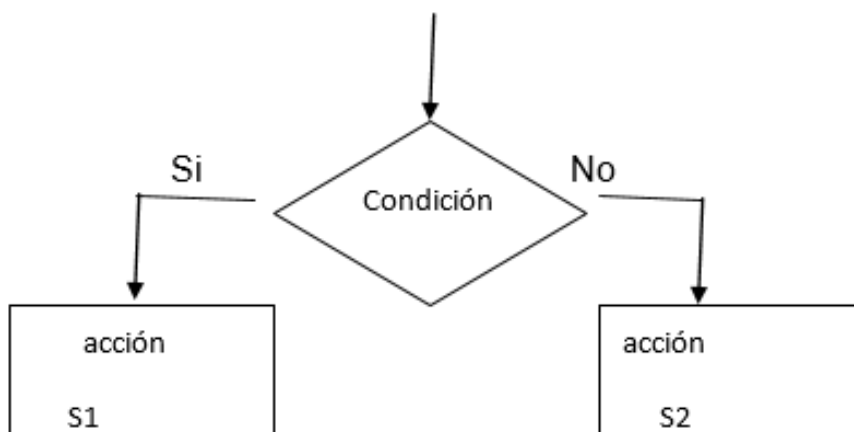


Condicionales

Estructura Condicionales

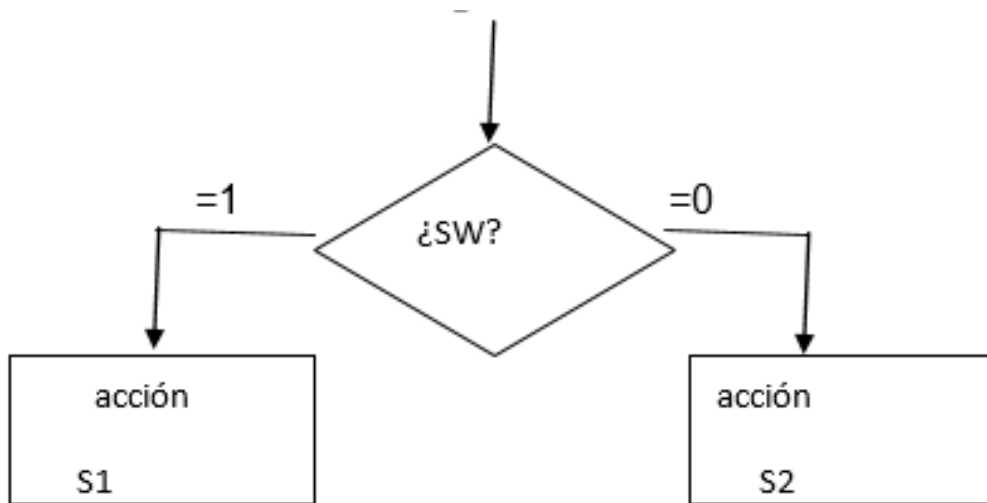
Estructuras de decisión o selección

Cuando se quiere especificar dos o más caminos alternativos en un algoritmo se deben utilizar estructuras de decisión o selección. Una instrucción de decisión o selección evalúa una condición y en función del resultado de esa condición se bifurca en un determinado punto.



Interruptores

Un interruptor o conmutador (switch) a veces se les denomina indicador, o bandera (flag) es una variable que puede tomar diversos valores a lo largo de la ejecución del programa y que permite comunicar información de una parte a otra del mismo. Los interruptores pueden tomar dos valores diferentes 1 y 0 (De ahí su nombre interruptor prendido apagado).



Programación

Desde que se desarrollaron las máquinas programables se han desarrollado lenguajes con los cuales las personas puedan dar órdenes a éstas. En su orden los lenguajes de programación se pueden clasificar así:

- **Lenguaje de máquina:** Las primeras computadoras se programaban en código de máquina. Se puede decir que los programas eran diseñados en código binario. Eran difíciles de leer, difíciles de entender y por su puesto difíciles de corregir. Los programas se caracterizaban por ser pequeños.
- **Lenguajes de Bajo Nivel:** Para dar solución a lo difícil que era programar en código máquina, se desarrolló un lenguaje conocido como lenguaje ensamblador. Este lenguaje era encargado de tomar algunas palabras comunes a una persona y traducirlas al código máquina. Lo anterior facilitaría un poco la escritura de programas.
- **Lenguajes de alto nivel:** Como las personas resuelven problemas y se comunican en lenguajes naturales (español, inglés, francés, etc.), se desarrollaron lenguajes de programación que estuvieran mas cerca de ésta manera de resolver problemas. Como se hace necesario traducir el programa a lenguaje de máquina, en los lenguajes de alto nivel esa operación la realiza algo que se conoce con el nombre de Compilador.

Tipos de programación

Dependiendo del lenguaje de programación que se elija, se puede hablar del tipo de programación que se va a realizar.

-**Secuencial:** Se considera programación secuencial a los programas que se diseñan con instrucciones que van unas detrás de otras. Las líneas se ejecutan una a una en secuencia.

-**Estructurada**: Se considera programación estructurada a la programación que se hace por módulos. Cada módulo realiza alguna tarea específica y cuando se necesite esa tarea simplemente se hace el llamado a ese módulo independiente de que se tengan que ejecutar los demás.

-**Orientada a Objetos**: Se considera programación orientada a objetos aquellos lenguajes que permiten la utilización de objetos dentro del diseño del programa y el usuario puede pegar a cada objeto código de programa.

- **Lógica o de lenguaje natural**: son aquellos programas que se diseñan con interfaces tal que la persona o usuario puede ordenar a la máquina tareas en un lenguaje natural. Pueden interactuar como una persona pero nunca llegan a producir conocimiento. Estos lenguajes se desarrollaron con base en las estructuras de sus antecesores. Recorren o navegan las bases de datos obedeciendo a reglas.

-**Inteligencia Artificial**: Los programas de inteligencia artificial Son programas que se acercan a la inteligencia humana. Estos programas son capaces de desarrollar conocimiento. Este tipo de lenguajes trabajan similar a la mente humana.

Programación Estructurada

Programación Estructurada es una técnica en la cual la estructura de un programa, esto es, la interrelación de sus partes realiza tan claramente como es posible mediante el uso de tres estructuras lógicas de control:

- Secuencia: Sucesión simple de dos o mas operaciones.
- Selección: bifurcación condicional de una o mas operaciones.
- Interacción: Repetición de una operación mientras se cumple una condición.

Estructura secuencial o secuencia de instrucción

Es aquella en que una acción sigue la secuencia. La acción B se ejecuta después de la A y ninguna acción puede ejecutarse entre ellas. La acción C, sigue a la acción B, y así sucesivamente.

La estructura secuencial o secuencia de instrucción



A,B representan las instrucciones que pueden ser de lectura, asignación o impresión. Las líneas verticales; significan que pueden existir mas instrucciones. Las estructuras selectivas se utilizan para tomar decisiones lógicas; de ahí que se suelen denominar también estructuras de decisión o alternativas. En las estructuras selectivas se evalúa una condición y en función del resultado de la misma se escoge una opción u otra. Las condiciones se especifican usando expresiones lógicas. La representación de una estructura selectiva se hace con palabras en pseudocódigo (if, then, else o bien en español si, entonces, sino), con una figura geométrica en forma de rombo o bien con un triángulo en el interior de una caja rectangular. Las estructuras lógicas selectivas se encuentran

en la solución algorítmica de casi todo tipo de problemas. Se utilizan cuando en el desarrollo de una solución de un problema se debe tomar una decisión para establecer un proceso o señalar un camino alternativo a seguir.

CLASES

Simples

Compuestas

Múltiples

ESTRUCTURAS SELECTIVAS

Anidadas

Operaciones o instrucciones de asignación

Se utilizan para asignar valores o expresiones a una variable. La asignación es una operación destructiva, esto significa que si la variable tenía asignado un valor anteriormente, este se destruye, conservando ahora el nuevo valor.

Normas para la creación de operaciones o instrucciones de asignación.

- El primer término debe ser siempre una variable.
- El valor del segundo término (2) es asignado al primer término o variable (1) - El símbolo utilizado para indicar asignación es el =(igual a).

Pasos para la solución de problemas.

Los pasos para la solución de un problema de computador son:

1. Análisis de la solución
2. Definición de variables
3. Diseñar diagrama de Flujo
4. Prueba de Escritorio.
5. Escribir Algoritmo.

1. Leer el enunciado (texto) hasta entender que está pidiendo el problema
 2. Escribir la definición de variables, este es muy importante, ya que ayuda a buscar la solución al problema planteado.
 2. Para la definición de variables se debe tener en cuenta lo siguiente:
-

- Asignar el nombre a cada una de las variables que se necesitan para resolver el problema.
 - Al lado de cada una de las variables, hacer una descripción del uso(s) que se le va a dar a dicha variable. Ejemplo COD= sirve para leer el código de un estudiante.
 - Cuando se está en proceso de definir una variable, se debe tener presente, para que no se olvide escribir ninguna, es conveniente organizarlas en el siguiente orden. Entrada, proceso, salida. (E,P,S).
 - Las variables de entrada son aquellas que sirven para leer o asignar un valor inicial con el cual se puede comenzar una acción o proceso.
 - Las variables de proceso, son aquellas que se utilizan para calcular y guardar un valor después de afectada la operación que se necesite. - Las variables de entrada pueden ser también de salida.
 - Las variables de proceso pueden también ser de salida
 - Las variables de entrada se pueden también utilizar como de proceso.
 - En la definición de variables se deben indicar todos los usos que se le van a dar a las variables.
3. Construir el Diagrama de Flujo: que es la representación gráfica de una serie de pasos ordenados y lógicos, que llevan a la solución de un problema, a la realización de una actividad o tarea o a la obtención de una respuesta.

4 Practicar una prueba de escritorio. Esta prueba es manual por eso se llama de escritorio, porque normalmente se usa papel y lápiz para realizarla. El objetivo de esta prueba es determinar si la solución dada en ese momento, a través de un diagrama, es la solicitada en el enunciado del ejercicio. Para ello se debe hacer lo siguiente:

- Sacar en forma horizontal y en orden, todas las variables utilizadas
- Recorrer paso a paso el diagrama e ir indicando todo lo registrado debajo de las variables correspondientes. Si la prueba indica que los resultados no son los esperados, se debe volver a desarrollar el diagrama.

5. Escribir el algoritmo correspondiente. Ejemplo. Escribir un algoritmo que calcule e imprima el valor de C.

```

Inicio
  Lea A,B
  Calcula C= A+B
  Escriba C
Fin-algoritmo

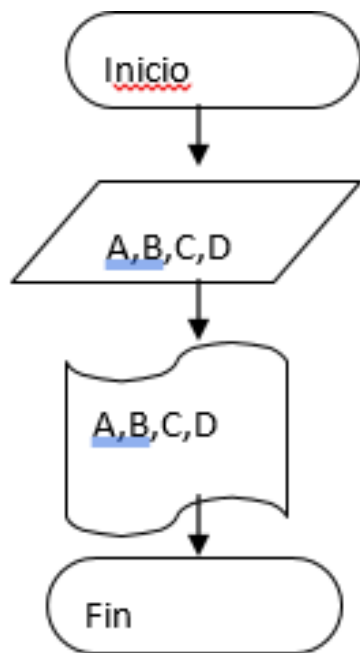
```

Ejemplo

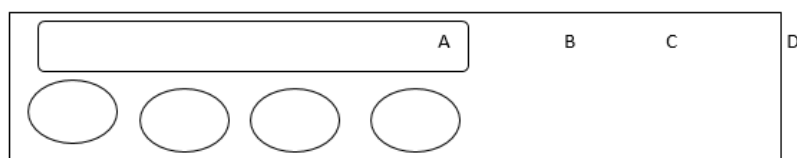
1. Leer 4 números y luego imprimirlos

- Análisis de la solución: en este caso es necesario definir 4 variables, cada una para leer un número.
- Definición de las variables: A, se utiliza para leer el primer número y posteriormente visualizarlo : B, se utiliza para leer el segundo número y posteriormente visualizarlo : C, se utiliza para leer el tercer número y posteriormente visualizarlo: D, se utiliza para leer el cuarto número y posteriormente visualizarlo.

Diagrama de Flujo



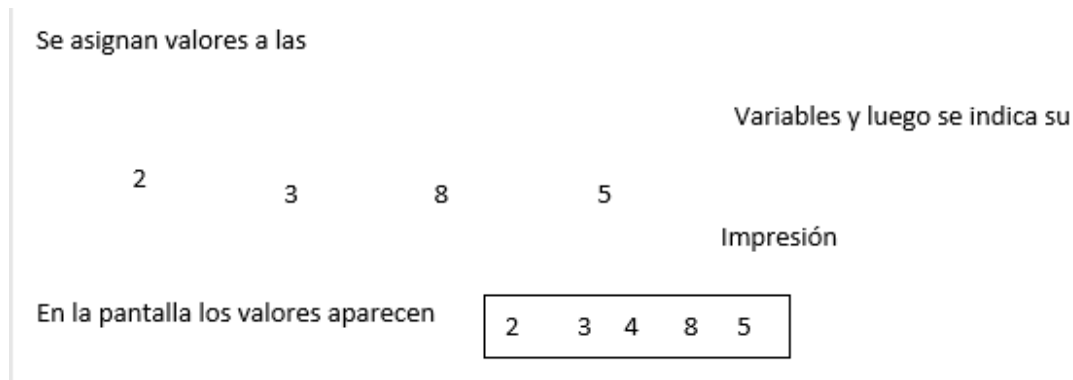
Prueba de escritorio:



C

A

B



Cuando se compara lo encontrado en la prueba de escritorio, con el enunciado se cumplen las 2 cosa. Que se leen los datos. Se imprimen o muestra los valores guardados en las variables, por lo tanto se puede continuar.

- e. Escribir el algoritmo: si el diagrama funciona correctamente solo, queda interpretarlo y transformarlo.

Inicio lea A;B;C;D imprima A;B;C;D
Fin_algoritmo

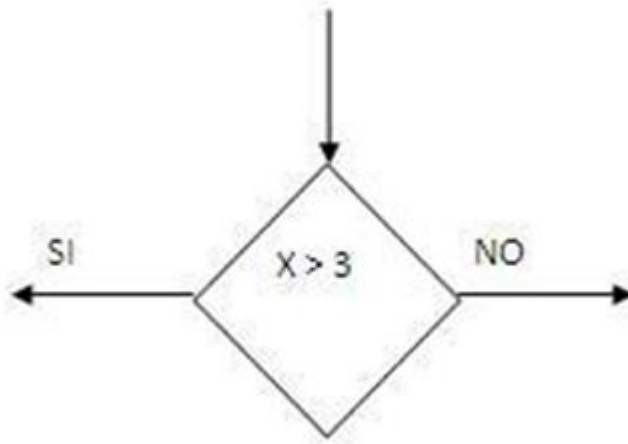
2.. Desarrolle un algoritmo que le permita calcular el área (A) de un segmento de círculo. Análisis: Para calcular el área de un segmento de círculo lo primero que hay que hacer es leer el valor del radio del vinculo y leer el valor de X que es la distancia del centro al segmento. Una vez leído dichos valores se calcula aplicando la fórmula respectiva y por último se escribe el valor del área.

Leer (X y r)
Calcular (Area)
Escribir

Seudo lenguaje
Inicio
Leer el valor de X y r
Calcular el valor de área
Escribir el valor de Area (A)
Fin

Estructura de Selección o de Instrucción

Las operaciones o instrucciones de decisión trabajan con las estructuras de selección o instrucciones y sirven para la solución de casi todo tipo de problemas. Se utilizan cuando en el desarrollo de la solución de un problema se debe tomar decisión, con el fin de establecer que camino seguir. El símbolo para indicar selección o pregunta es el rombo. Mínimo debe tener 2 salidas



En las estructuras de selección de instrucciones cuando la condición se cumple tomara el camino SI o Verdadero; si la condición no se cumple, se ira por el camino NO o Falso (si no) en cualquiera de los casos después de ejecutar las instrucciones que se encuentren en el respectivo camino, saldrá de la estructura e ira a ejecutar las instrucciones que se encuentran por fuera y a continuación de ella, hasta encontrar el fin del proceso.

Las operaciones o instrucciones de decisión se trabajan en las estructuras de de selección de instrucción llamadas comúnmente.

Estructura selectiva simple o compuesta.

- La estructura selectiva lógica "si entonces" permite que el flujo siga por un camino específico si se cumple una condición o un conjunto de condiciones.
- Si al evaluar la condición (o condiciones) el resultado es verdadero, entonces se ejecuta (n) cierta (s) operación (es), luego continua con la secuencia normal del proceso.
- Se utiliza cuando alguna operación está condicionada para que se lleve a cabo, pero no tiene una opción alterna

formato:

si condición entonces operación (es) { Fin del condicional } Donde:

si.- identifica la estructura selectiva

condición.- expresa la condición o conjunto de condiciones a evaluar

entonces.- indica el curso de acción a seguir si se cumple la condición

Operación.- expresa la operación o conjunto de operaciones.

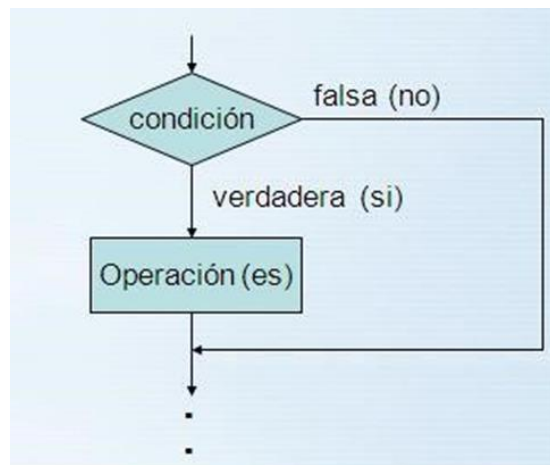
fin del condicional.- indica el fin de la estructura de selección (si)

Funcionamiento:

Al llegar al si se evalúa la condición (es):

a). Si se cumple, se ejecuta (n) la (s) operación (es) del entonces y luego salta hasta el siguiente paso después del fin del condicional.

b). Si no se cumple, salta hasta después del fin del condicional, es decir no hace nada.



Ejemplo.

Leer la calificación o nata de un estudiante; si la nota es mayor que 3, escribir o imprimir un mensaje que diga Aprobado, si no terminar.

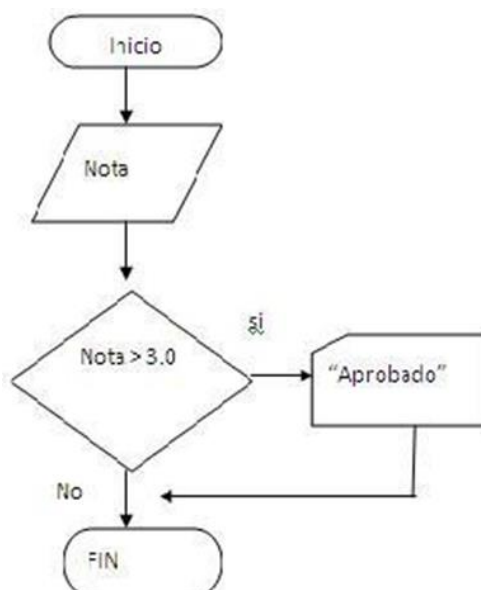
a. Análisis de la solución

El ejercicio pide que leamos una nota; que después se pregunte si es mayor que 3 y dependiendo de eso, si resulta ser > 3 imprimir el mensaje que nos indique el enunciado.

b. Definición de Variables

Solo se necesita una variable para leer la nota.

c. Diagrama de flujo



d. Prueba de escritorio

Nota	3.5	"aprobado"	Cuando la condición se cumple	imprime el Aprobado
	2.8		cuando la condición no se cumple	va a buscar el fin de la estructura.

Para este ejemplo se harán 2 pruebas de escritorio con notas aleatorias, de tal manera que se vaya primero por un camino y luego por otro.

Imagen17. Prueba de escritorio

e. Algoritmo

Inicio

Lea Nota

Si Nota > 3 entonces Imprima " Aprobado" Fin si

Fin algoritmo

La estructura Si entonces SI/No (estructura selectiva doble o compuesta. If – Then – Else)

Permite que el flujo del diagrama se bifurque por 2 caminos diferentes. Si al evaluar la condición el resultado es Si o verdadero, entonces se ejecutaran las instrucciones que se encuentran por dicho camino; si la condición no se cumple se ejecutaran las instrucciones que se encuentran por No (Si-No) o falso. Si al evaluar la condición (o condiciones) el resultado es verdadero, entonces sigue por un camino específico y se ejecuta (n) cierta (as) operación (es). Por otra parte, si el resultado es falso entonces se sigue por otro camino y se ejecuta (n) otra (s) operación (es). En ambos casos, luego de ejecutarse la (s) operación (es) indicada (s), se continúa con la secuencia normal del proceso. Por la naturaleza de éstas, se debe ejecutar una o la otra, pero no ambas a la vez, es decir, son mutuamente excluyentes.

Formato:

si condición

entonces

hacer operación 1

sino

hacer operación 2

{ Fin del condicional } Donde:

si.- identifica la estructura selectiva

condición.- expresa la condición o conjunto de condiciones a evaluar

entonces.- indica el curso de acción a seguir si se cumple la condición.

Operación 1.- expresa la operación o conjunto de operaciones.

Operación 2.- expresa la operación o conjunto de operaciones.

fin del condicional.- indica el fin de la estructura de selección (si)

Funcionamiento:

Al llegar al si se evalúa la condición (es):

a). Opción verdadera (entonces). si se cumple, se ejecuta (n) la (s) operación (es) del entonces y luego salta hasta el siguiente paso después del fin del condicional.

b). Opción falsa (sino). De lo contrario, salta hacia el sino, ejecuta la (s) operación (es), y después salta hasta el siguiente paso después del fin del condicional.

Ejercicio

Después de leer 3 notas y calcular el promedio, determinar o imprimir si un estudiante aprueba o reprueba una asignatura. Sabiendo que aprobará si el promedio es ≥ 3.0 si no reprobará.

a.. Análisis de la solución. Se deben definir tres variables cada una para leer una nota y

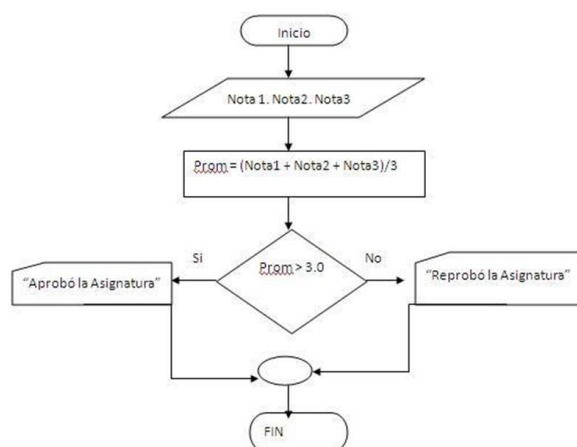
luego definir la variable que guardara el promedio de dichas notas; luego se sabe si el promedio es ≥ 3.0 o inferior.

b. Definición de variables

Nota1, Nota2 Nota 3 = Sirven para leer cada una de las notas de la asignatura.

Prom = sirve para calcular y guardar le promedio de las notas leídas.

c. Diagrama de Flujo.



d. prueba de escritorio

Nota 1	Nota 2	Nota3	Prom	
3.5	4.5	2.7	3.43	"Aprobó la asignatura"
2.7	3.2	2.3	2.73	"Reprobó la asignatura"

e.. Algoritmo

Inicio

Lea Nota1, Nota2 Nota3

Prom = (Nota1 + Nota2 + Nota3) /3 Si Prom > = 3.0 entonces

Imprima " Aprobó la asignatura"

Si no

Imprima " Reprobó la asignatura"

Fin si

Fin algoritmo

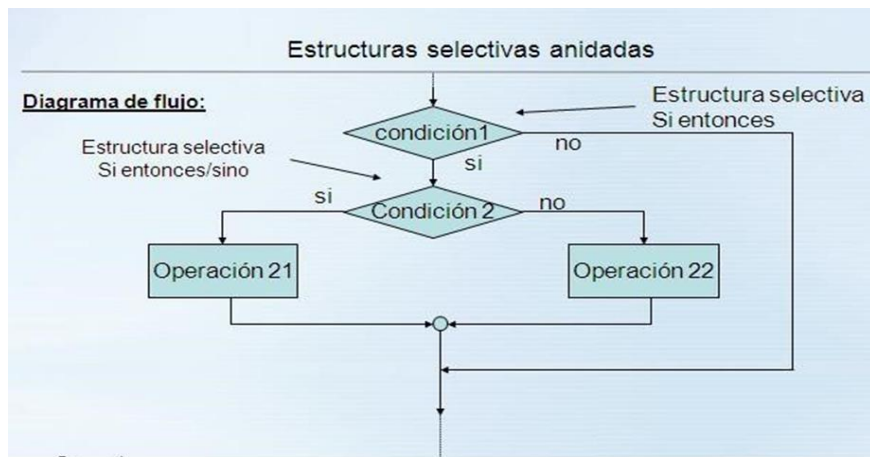
Estructura selectiva anidada

Existen numerosos casos en el desarrollo de la solución de problemas en el que luego de tomar una decisión y marcar el camino correspondiente a seguir, es necesario tomar otra decisión.. Se señala, luego de evaluar las condiciones, la rama correspondiente a seguir, y nuevamente podemos tener que tomar otra decisión. El proceso puede repetirse numerosas veces. En este caso, para resolver el problema, estamos aplicando estructuras selectivas en cascada o anidadas. Algunos ejemplos son:

```

n      si condición1 entonces
        n1      si condición2  entonces
            hacer operación21
            sino
                hacer operación22
        n2
    fin del condicional del paso n1
n + 1 fin del condicional del paso n

```



Ejemplo: desarrollar un algoritmo y diagrama de flujo el cual, dado un valor entero, determinar si es cero, positivo o negativo.

Algoritmo Dado un valor entero determinar si es cero, positivo o negativo

inicio

Declaración de variables: num: entero

Leer num si num == 0 entonces

imprimir "cero" sino

si num > 0 entonces

imprimir "positivo" sino

imprimir "negativo" fin del condicional paso 4.1

Fin del condicional del paso 4

