

Capítulo 3

Procesos con estructuras de selección

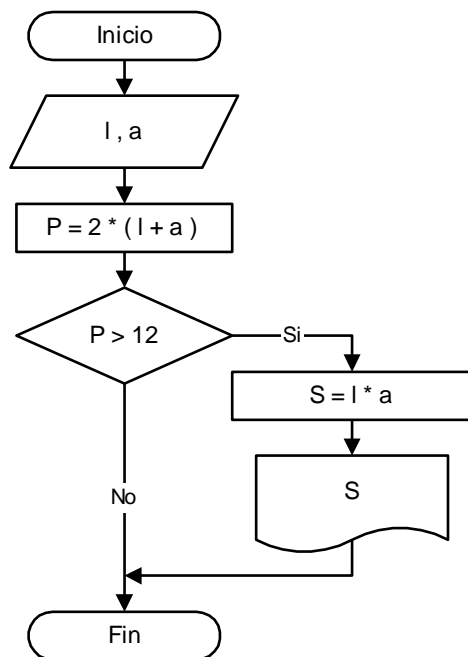
Procesos con estructuras de selección

Los procesos lineales vistos en el capítulo anterior permiten solucionar un muy reducido número de problemas, ya que cualquier pequeña complejidad incorporada nos obliga a tomar alguna alternativa de entre varias posibles.

Para ello la programación provee varias estructuras de selección, de las cuales estudiaremos la más simple: la decisión con dos ramas de salida, que a pesar de su simplicidad nos permite resolver un numeroso grupo de problemas.

Ejercicio 3.1:

Efectuar el diagrama de flujo de un programa que lea el largo l y el ancho a de un rectángulo, y calcule su perímetro P . Si el perímetro P es mayor a 12, que calcule e imprima su superficie S .



Fórmulas:

Perímetro: $P = 2 \cdot (l + a)$

Superficie: $S = l \cdot a$

Prueba de escritorio 1:

l	a	P	S
3	2	10	

Salida por pantalla 1

Prueba de escritorio 2:

l	a	P	S
5	7	24	

24

35

Salida por pantalla 2

35

En este ejercicio nos encontramos con la primera decisión, donde se compara el perímetro P con un valor 12. La condición del diagrama debe ser formulada de manera que pueda tener solamente dos respuestas (Si o No, Verdadero o Falso, etc.) para poder utilizar el gráfico de decisión visto en el capítulo 1.

En las dos ramas de salida del gráfico de decisión pueden existir procesos, y en algunos casos los procesos se encontrarán en una sola de las ramas. Para esta última situación se sugiere formular la condición de manera que los procesos queden ubicados en la rama del SI, dejando vacía la rama de salida del NO. Esto obedece a facilitar su codificación con la estructura cerrada del IF-THEN como se verá más adelante.

Una vez efectuados los procesos de cada rama, las líneas de flujo vuelven a unirse más abajo, en correspondencia con el cierre de la estructura de decisión, y a partir de este cierre continúa una sola línea de flujo.

En el Ejercicio 3.1 el cierre de la estructura condicional conduce a la finalización del diagrama de flujo al cumplir lo solicitado por el enunciado.

Operadores relacionales

Los operadores relacionales comparan dos datos del mismo tipo y producen un valor lógico como resultado de la comparación, es decir, son los que producen la respuesta Verdadero o Falso Si o No. En el Ejercicio 3.1 se utilizó el operador relacional > (mayor que), pero se admiten una cantidad de seis operadores de relación, como lo muestra el cuadro siguiente:

Operador de relación	Significado
<	Menor que
>	Mayor que
=	Igual a
<=	Menor o igual que
>=	Mayor o igual que
<>	Diferente a

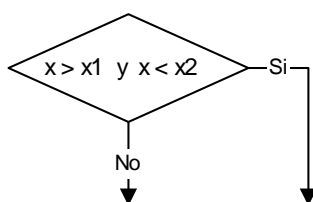
Unión de varias condiciones en un mismo gráfico de decisión

El gráfico de decisión estudiado tiene solo dos salidas, por Si o por No, de manera que la pregunta o condición que se formula en su interior debe tener la estructura adecuada que brinde solo dos respuestas posibles.

En muchos casos se necesita hacer más de una comparación para tomar una decisión en el flujo del proceso. Estas preguntas simultáneas deben ser combinadas mediante operadores lógicos que funcionan de manera idéntica al álgebra de Boole, ya estudiado en materias anteriores.

En el desarrollo del curso es suficiente el aprendizaje de dos operadores lógicos: Y (AND) y O (OR). Los ejemplos siguientes aclaran el uso de estos operadores.

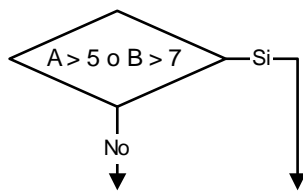
Operador Y (And)



Al igual que en álgebra, la salida por Si se producirá en el caso de que todas las condiciones resulten verdaderas. Es suficiente que una de las condiciones no se cumpla para que el flujo tome la rama del No.

If (x > x1) And (x < x2) Then

Operador O (Or)

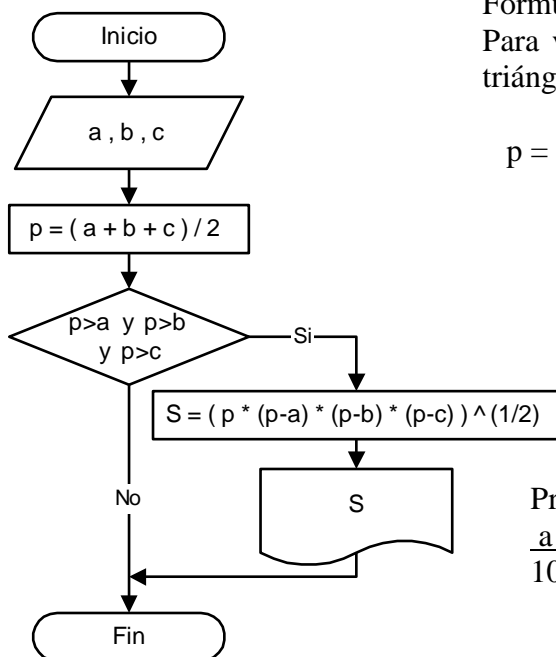


En este caso es suficiente que alguna de las condiciones sea verdadera para obtener una salida por la rama del Si. Si todas las condiciones fueran falsas, el flujo continúa por la rama del No.

If (A > 5) Or (B < 7) Then

Ejercicio 3.2:

Efectuar el diagrama de flujo de un programa que lea tres valores a, b y c, determine si estos valores pueden ser los lados de un triángulo, y en caso afirmativo que calcule la superficie S del triángulo y la imprima.



Fórmulas:

Para verificar si los valores pertenecen a los lados de un triángulo utilizamos la siguiente propiedad:

$$p = \frac{a + b + c}{2} \quad \text{debe ser mayor a todos y a cada uno de los lados}$$

es decir: $p > a$ y $p > b$ y $p > c$

La superficie se calcula con la siguiente fórmula:

$$S = \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)}$$

Prueba de escritorio 1:

a	b	c	p	S
10	8	6	12	24

Salida por pantalla 1

24

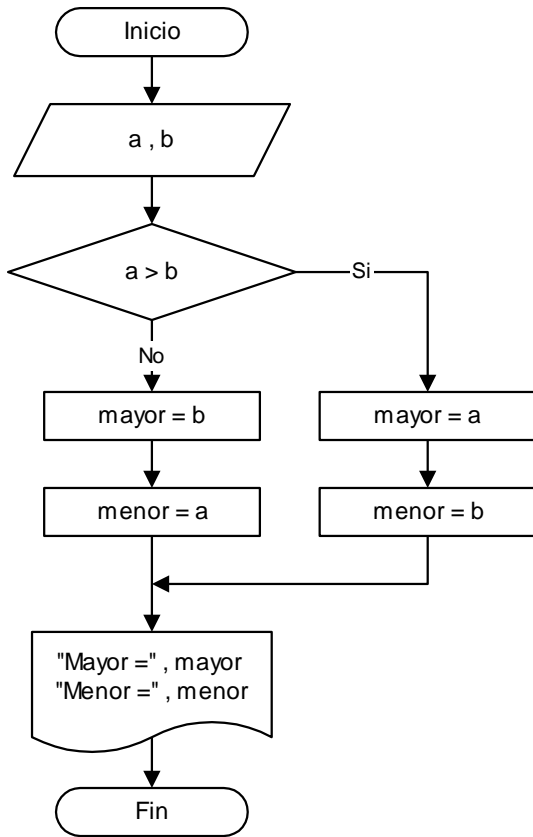
Prueba de escritorio 2:

a	b	c	p	S
10	2	4		8

Salida por pantalla 2

Ejercicio 3.3:

Efectuar el diagrama de flujo de un programa que lea dos números a y b, los compare y almacene en dos variables diferentes el mayor y el menor de ellos. Imprimir los resultados.



Prueba de escritorio:

a	b	mayor	menor
10	12	12	10

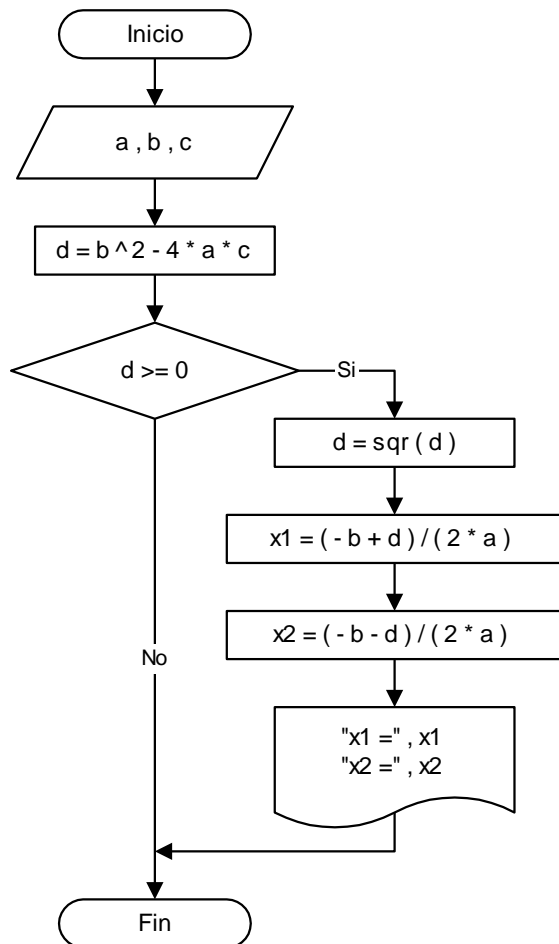
Salida por pantalla

```

Mayor = 12
Menor = 10
  
```

Ejercicio 3.4:

Efectuar el diagrama de flujo de un programa que calcule las raíces x_1 y x_2 de una ecuación de 2º grado: $a \cdot x^2 + b \cdot x + c = 0$, considerando además la posibilidad de raíces imaginarias



Fórmula:

$$x_1 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x_2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Prueba de escritorio:

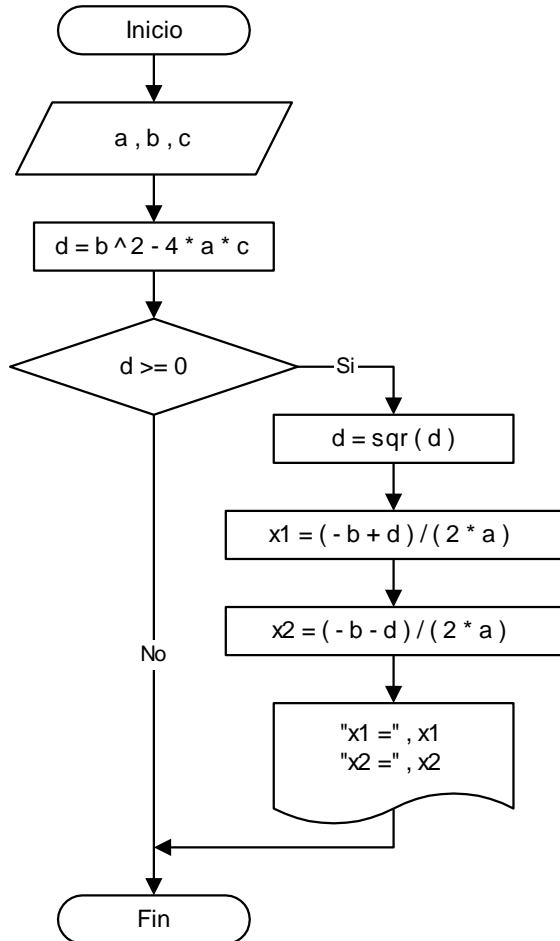
a	b	c	d	x1	x2
2	10	3			
			76		
			8.717798		
				-0.32055	-4.67945

Salida por pantalla

```
x1 = - 0.32055
x2 = - 4.67945
```

Ejercicio 3.5:

Efectuar el diagrama de flujo de un programa que calcule las raíces x_1 y x_2 de una ecuación de 2º grado: $a \cdot x^2 + b \cdot x + c = 0$.



Fórmula:

$$x_1 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$$x_2 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Prueba de escritorio:

a	b	c	d	x1r	x1i	x2r	x2i
4	4	2					
			-16				
			4				
				-0.5			
					0.5		
						-0.5	
							-0.5

Salida por pantalla

```

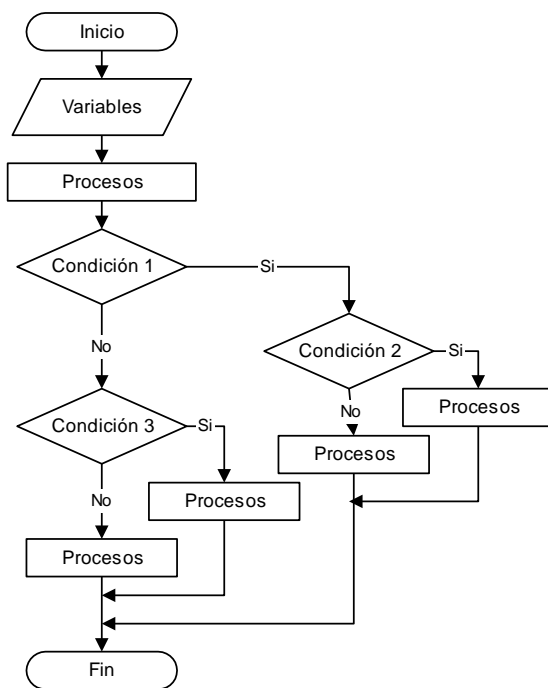
x1 = - 0.5 + 0.5 i
x2 = - 0.5 - 0.5 i
    
```

Estructuras de selección anidadas

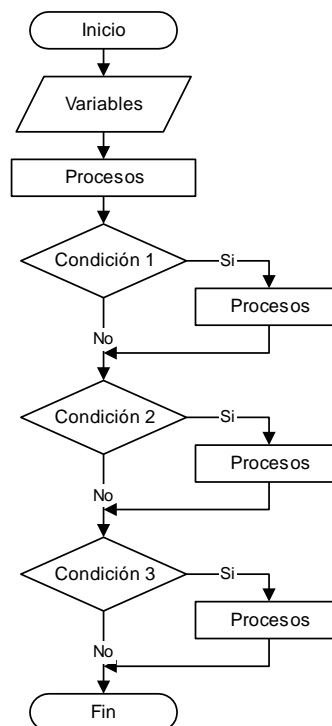
Además de las estructuras de selección simples que hemos visto, algunas de las cuales tienen procedimientos en la rama del SI y otras tienen procedimientos en las dos ramas condicionales, se pueden presentar casos en que se requiera más de una condición, que no se pueda implementar con conectores lógicos, o también varias condiciones sucesivas, que nos obliguen a utilizar estructuras condicionales anidadas.

En estas soluciones se debe tener la precaución de no cruzar líneas de flujo de las ramas afirmativas y negativas de las diferentes condiciones, de tal manera que sean claras sus trayectorias y sus puntos de cierre, para cada una de las condiciones, lo cual tendrá a su vez el resultado de una codificación más clara y más sencilla de implementar.

Como ejemplos esquemáticos se pueden presentar los siguientes diagramas de flujo correctos:



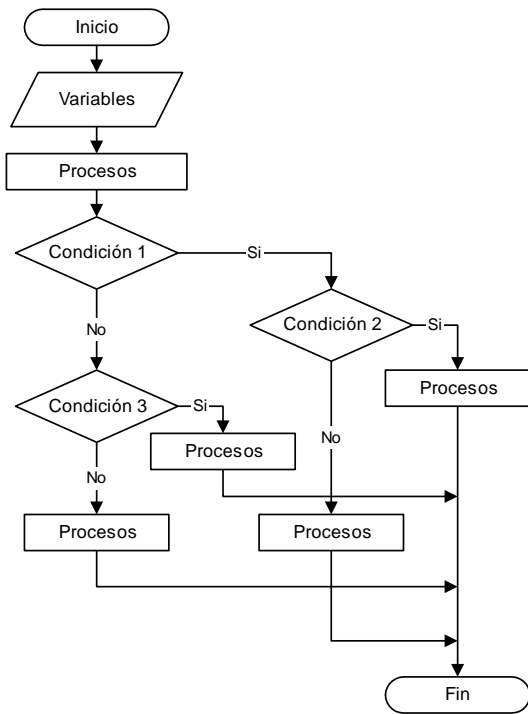
Ejemplo Correcto



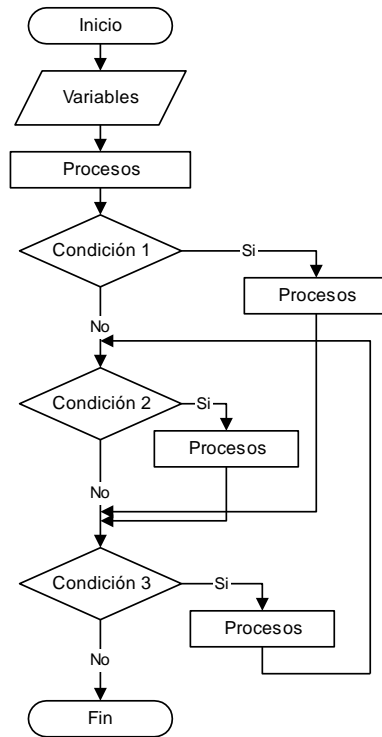
Ejemplo Correcto

Al analizar las líneas de flujo de las ramas afirmativas y las ramas negativas no se notan cruces de líneas de flujo. Existen numerosas variantes de anidar Estructuras Condicionales, o sea que no sería posible esquematizar todos los casos.

Podemos presentar algunos ejemplos **incorrectos** para que quede con mayor claridad lo que se debe evitar:

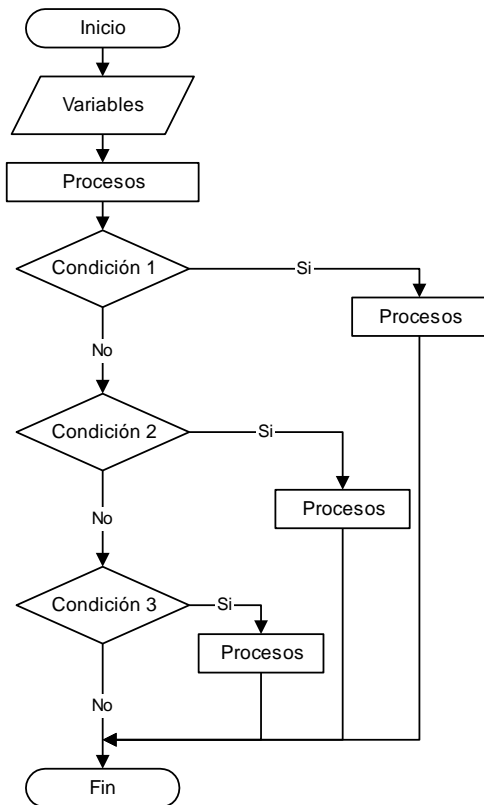


Ejemplo Incorrecto



Ejemplo Incorrecto

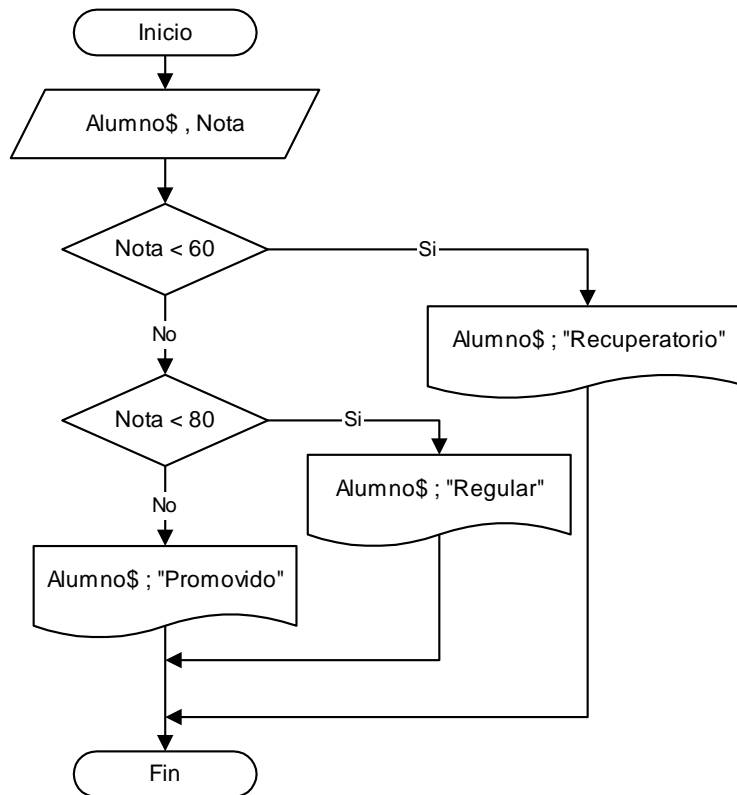
Cada estructura condicional debe tener su punto de cierre identificado, independiente de las demás estructuras condicionales por lo cual el siguiente ejemplo también es incorrecto.



Ejemplo Incorrecto

Ejercicio 3.6:

Efectuar el diagrama de flujo de un programa que permita ingresar el nombre de un Alumno y la Nota obtenida en un parcial, y en base a ello defina si su situación es Promovido, Regular o bien debe rendir Recuperatorio en caso de obtener una nota menor a 60 pts. Imprimir el nombre del alumno y el resultado obtenido.



Ejercicios Adicionales Capítulo 3:

1. Realizar el diagrama de flujo de un programa que lea dos números a y b, e imprima el mayor de ellos.
2. Realizar el diagrama de flujo de un programa que permita leer cuatro números y a continuación imprima el mayor de los cuatro.
3. Realizar el diagrama de flujo de un programa que lea tres números y encuentre si alguno de ellos es igual a la suma de los otros dos. En ese caso, imprimir el número encontrado.
4. Realizar el diagrama de flujo de un programa que lea la nota numérica de un estudiante y asigne una calificación según la siguiente tabla:

90-100	Sobresaliente
80-89	Distinguido
70-79	Bueno
60-69	Aprobado
menos de 60	Insuficiente

Imprimir en pantalla la calificación asignada.

5. Realizar el diagrama de flujo de un programa que permita determinar si un número leído por teclado es divisible por 2, por 3 o por 5. Imprimir para cada caso la respuesta.
6. Realizar el diagrama de flujo de un programa que lea dos números A y B, los compare y almacene en dos variables diferentes el mayor y el menor de ellos. Imprimir los resultados.
7. Realizar el diagrama de flujo de un programa que permita determinar si un corredor será seleccionado para participar de una maratón cuyas condiciones son:
Para hombres menores a 40 años, el tiempo de calificación debe ser menor a 150 minutos.
Para hombres mayores a 40 años, el tiempo de calificación debe ser menor a 175 minutos.
Para mujeres, el tiempo de calificación debe ser menor a 180 minutos.
El proceso debe permitir ingresar desde el teclado los datos de edad, sexo y tiempo en la prueba de calificación.
8. Realizar el diagrama de flujo de un programa que permita resolver las raíces de la ecuación de segundo grado $y = a.x^2 + b.x + c$ contemplando todas sus variantes, controlando que el argumento de la raíz cuadrada no sea negativo, e imprimir las soluciones obtenidas.

$$x_1 = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$
$$x_2$$