

Estructura de decisi3n

Sitio: [Agencia de Aprendizaje a lo largo de la Vida](#)
Curso: Tecnicas de Programaci3n 1° G
Libro: Estructura de decisi3n

Imprimido por: Sebastian Puche
Día: domingo, 27 de octubre de 2024, 16:23

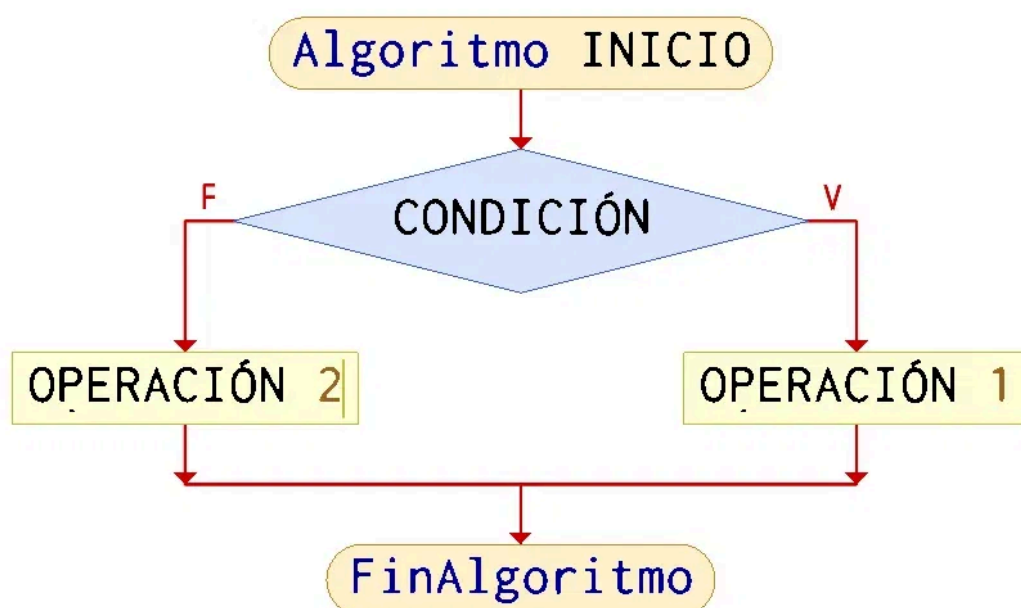
Tabla de contenidos

- 1. ¿Qué es una estructura de decisi3n?
- 1.1. ¿En qué se diferencia una estructura de secuencia de una estructura de decisi3n?
- 1.2. ¿C3mo se construye una estructura de decisi3n?
- 1.3. Estructuras de decisi3n anidadas
- 1.4. Estructura de decisi3n m3ltiple

1. ¿Qué es una estructura de decisión?



La mayor parte de la potencia del procesador proviene de la capacidad de cálculo, o sea de la capacidad de tomar decisiones y determinar qué acción tomar en el momento de la ejecución del algoritmo. La estructura de decisión consta de realizar una o una serie de acciones entre un conjunto de alternativas.



1.1. ¿En qué se diferencia una estructura de secuencia de una estructura de decisión?

La estructura de secuencia

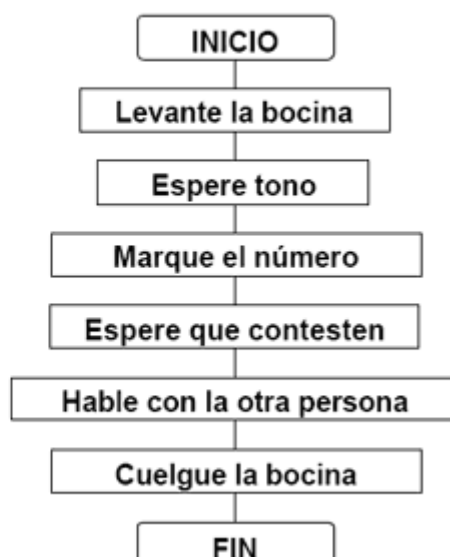
Sigue una serie de pasos predeterminados. Por ejemplo,

Pseudocódigo:

```

INICIO
  Levante la bocina
  Espere tono
  Marque el número
  Espere que contesten
  Hable con la otra persona
  Cuelgue la bocina
FIN
  
```

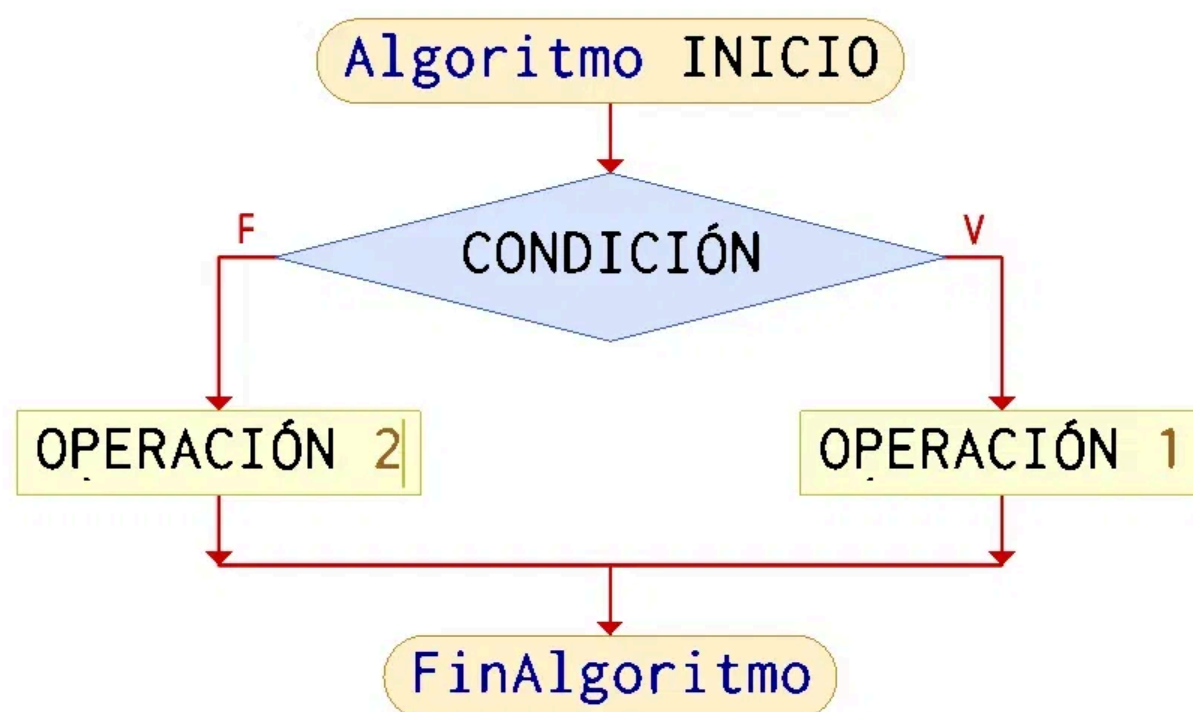
Diagrama de flujos:



La estructura de decisión

Debe elegir, a partir de los datos procesados, una alternativa de acción u otra un camino por el cual ejecutar unas sentencias u otras.

Por ejemplo,



1.2. ¿Cómo se construye una estructura de decisión?

La construcción comienza con la **palabra SI**, seguida de la **condición** que deba probarse (proposición lógica). La alternativa que debe ser ejecutada si la condición resulta **verdadera** se encuentra precedida por la palabra **ENTONCES**, la otra alternativa (**la falsa**) está precedida por la palabra **SINO**. Esta estructura finaliza con la palabra **FIN SI**.

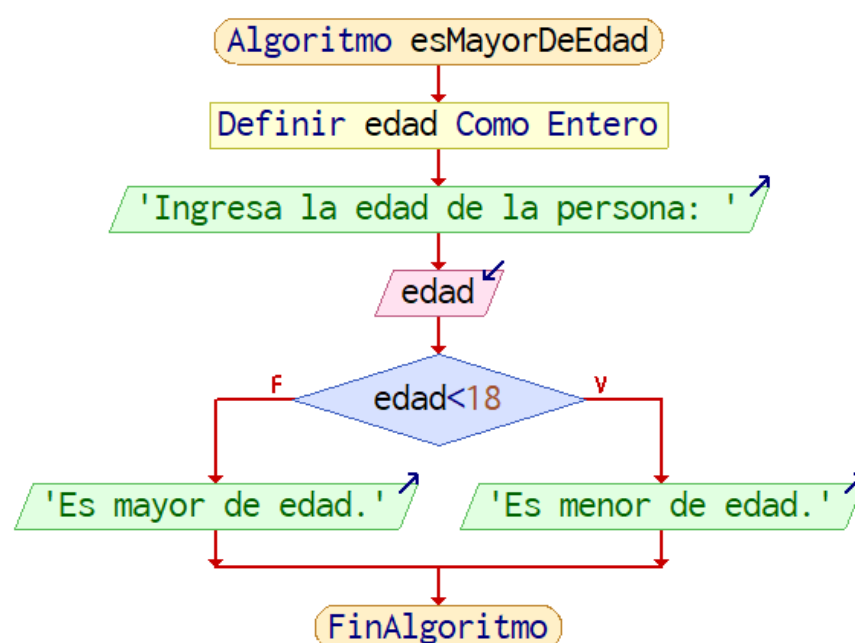


Como con toda estructura que veamos en el transcurso de la cursada, hacemos hincapié que solo debe tener un único punto de entrada y un único punto de salida. Tiene su entrada donde la condición es evaluada y la salida por fuera de ambas alternativas (centro).

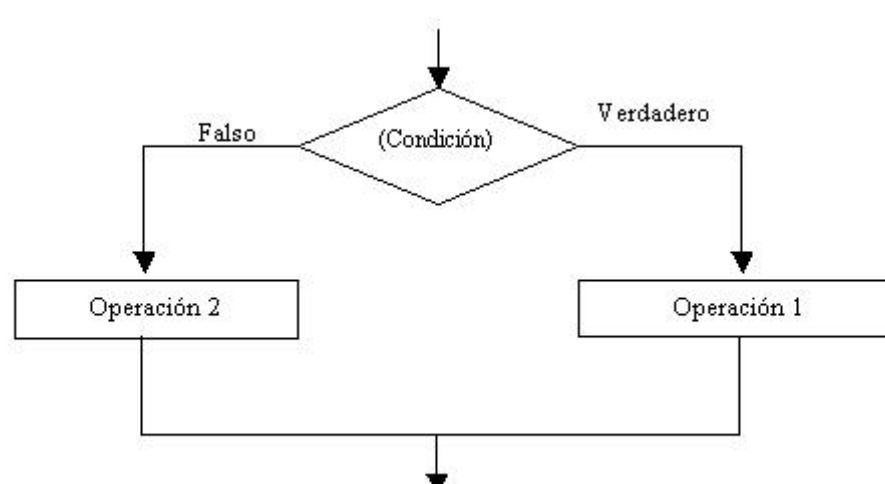


Veamos un ejemplo práctico:

- **Situación problemática:** solicitar el ingreso de un número entero que represente la edad de una persona e informar si dicha persona es mayor de edad o no.
- **Hipótesis:** asumimos mayor de edad a partir de los 18 años cumplidos.



Como vemos, se evalúa la condición y este nos arroja un valor verdadero o falso. En base a ese resultado de la condición, el algoritmo tomará un camino u otro.

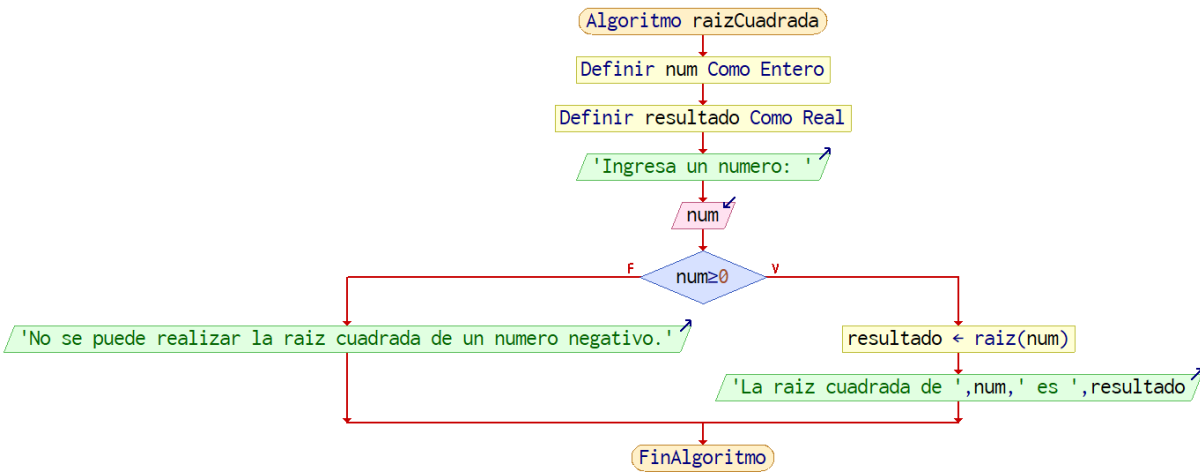




Veamos otro ejemplo.

Calcular la ra3z cuadrada de un n3mero, si este no es negativo, y en caso contrario escribir un mensaje al usuario aclaratorio.

Recordemos que no existe la ra3z cuadrada de un n3mero negativo porque ning3n n3mero al cuadrado es negativo.



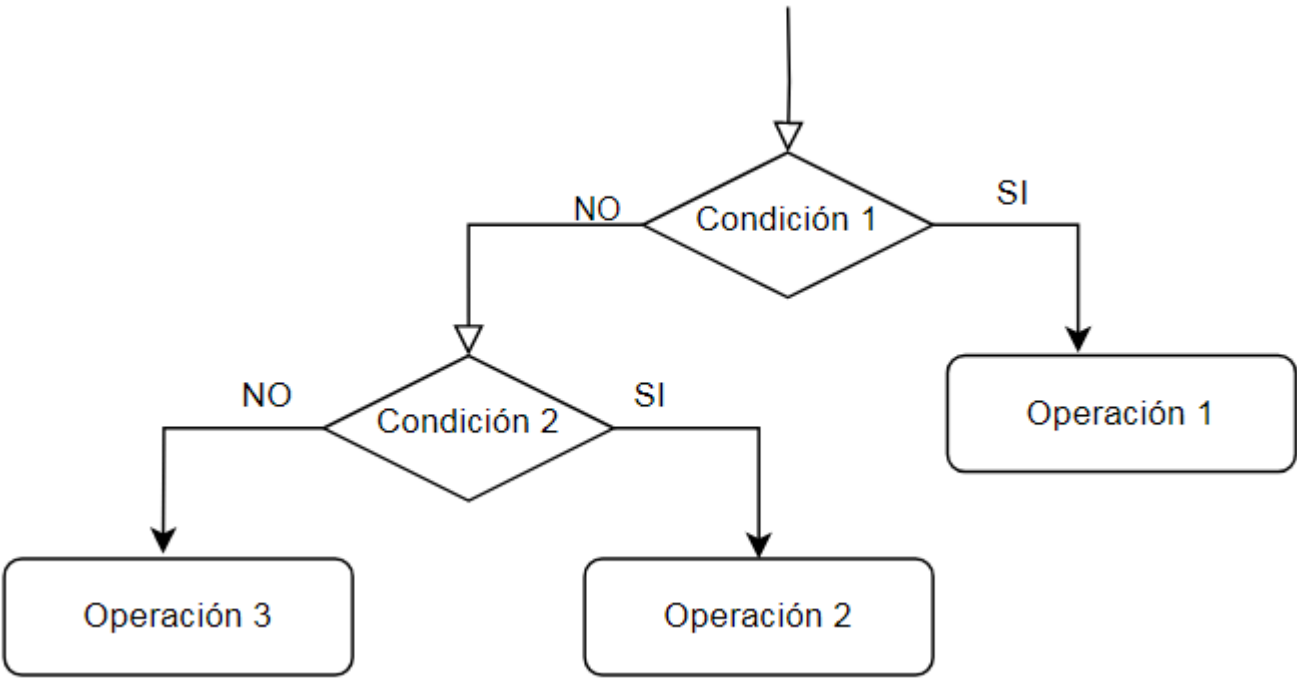
```
Algoritmo raizCuadrada
  Definir num Como Entero
  Definir resultado Como Real
  Escribir "Ingresa un numero: "
  Leer num
  Si num >= 0 Entonces
    resultado = raiz(num)
    Escribir "La raiz cuadrada de ", num, " es ", resultado
  SiNo
    Escribir "No se puede realizar la raiz cuadrada de un numero negativo."
  FinSi
FinAlgoritmo
```

Observemos que en la mayor3a de los lenguajes existe una funci3n para obtener la ra3z cuadrada de un n3mero que le indiquemos.

En el caso de PSeInt esa funci3n es ra3z ([numeroACalcular](#)).

1.3. Estructuras de decisi3n anidadas

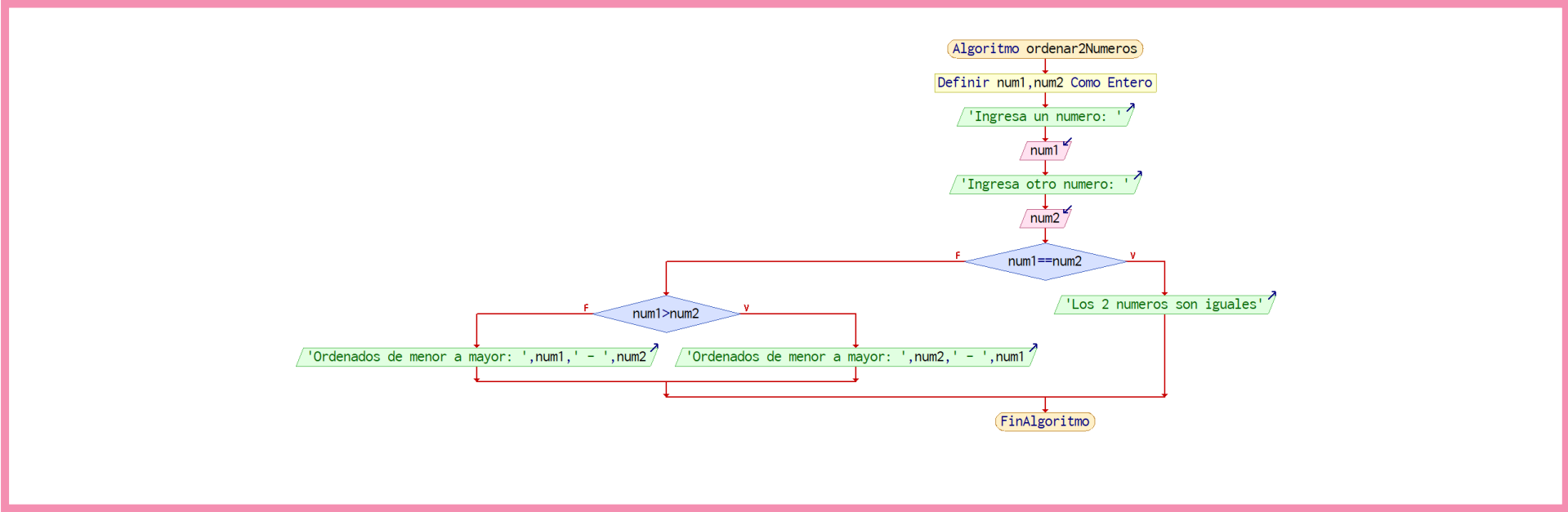
Tanto la alternativa verdadera como la falsa, pueden contener a su vez, estructuras de decisi3n (selecci3n), de secuencia o estructuras de repetic3n, dentro de ellas. A estas 3ltimas estructuras que est3n dentro de una alternativa previa de la estructura que las contiene, las llamamos estructuras de decisi3n anidadas.



 **Veamos un ejemplo:**

Realizar un algoritmo que ingrese dos n3meros enteros y los escriba ordenados de menor a mayor. Si son iguales escribir la leyenda “los 2 n3meros son iguales”.

```
Algoritmo ordenar2Numeros
  Definir num1,num2 Como Entero
  Escribir 'Ingresa un numero: '
  Leer num1
  Escribir 'Ingresa otro numero: '
  Leer num2
  Si num1 == num2 Entonces
    Escribir "Los 2 numeros son iguales"
  SiNo
    Si num1 > num2 Entonces
      Escribir "Ordenados de menor a mayor: ",num2, " - ", num1
    SiNo
      Escribir "Ordenados de menor a mayor: ",num1, " - ", num2
    FinSi
  FinSi
FinAlgoritmo
```





Otro ejemplo:

Ingresan un número entero que representa a uno de los días de la semana, los valores posibles que debería tomar son de 1 a 7. Se pide confeccionar un algoritmo que escriba el nombre del día de la semana dependiendo del valor de la variable día. Si día llegase a tomar un valor diferente, informar error.

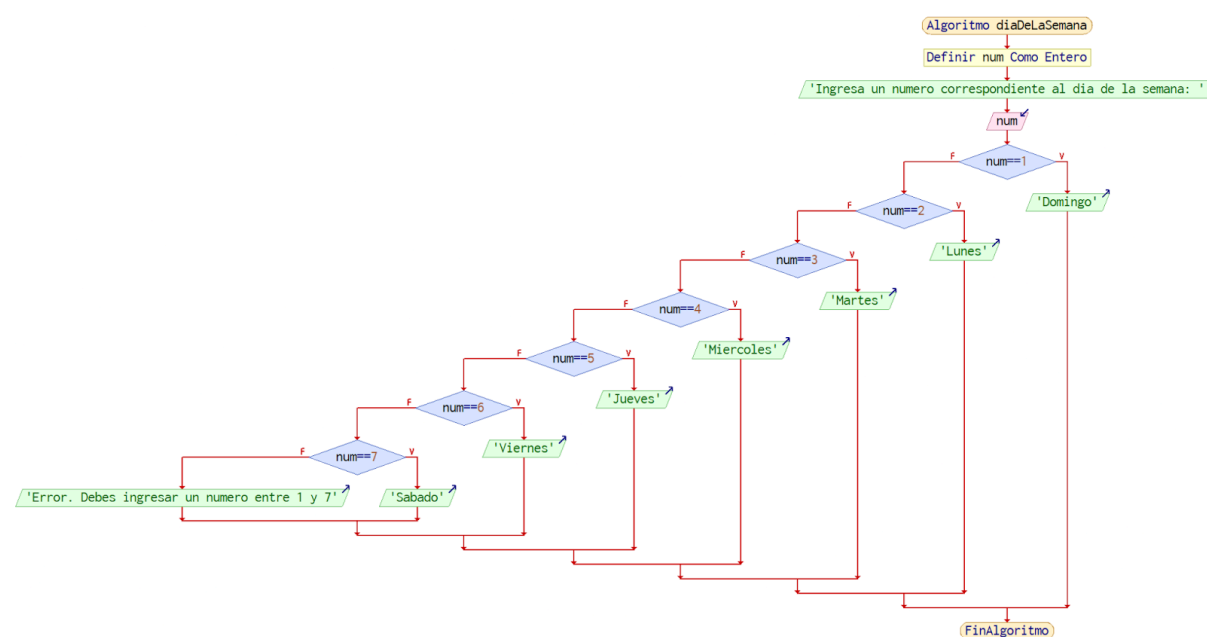


Hipótesis: asumimos como primer día de la semana el domingo.

```

Algoritmo diaDeLaSemana
Definir num Como Entero
Escribir 'Ingresa un numero correspondiente al dia de la semana: '
Leer num
Si num == 1 Entonces
    Escribir "Domingo"
SiNo
    Si num == 2 Entonces
        Escribir "Lunes"
    SiNo
        Si num == 3 Entonces
            Escribir "Martes"
        SiNo
            Si num == 4 Entonces
                Escribir "Miercoles"
            SiNo
                Si num == 5 Entonces
                    Escribir "Jueves"
                SiNo
                    Si num == 6 Entonces
                        Escribir "Viernes"
                    SiNo
                        Si num == 7 Entonces
                            Escribir "Sabado"
                        SiNo
                            Escribir "Error. Debes ingresar un numero entre 1 y 7"
                        FinSi
                    FinSi
                FinSi
            FinSi
        FinSi
    FinSi
FinSi
FinSi
FinSi
FinSi
FinSi
FinAlgoritmo

```



Como vemos en el gráfico anterior, resolver esto con la estructura de decisión simple de manera correcta implicaría bastante complejidad y se haría muy confuso. Para evitar este problema surge la estructura de decisión múltiple.

1.4. Estructura de decisi3n m3ltiple

En esta estructura no existe una **condici3n Si.. SINO**; lo que se hace es tomar la comparaci3n de una variable. El valor que pueda tomar esta variable podr3 estar o no dentro de las opciones de evaluaci3n de esta **estructura de decisi3n**.

La utilizamos cuando se necesita optar entre varias alternativas. Se usa para elegir un camino entre varios alternativos y funciona comparando una variable con una lista de constantes enteras o de car3cter. Cuando concuerda con alguna, se ejecuta la secuencia de sentencias asociada con esa constante.



Tomando el ejemplo de los d3as de la semana, lo representaremos con la estructura de decisi3n

m3ltiple. Notamos que el algoritmo es mucho m3s sencillo que la implementaci3n hecha con la estructura de decisi3n anidada.

```
Algoritmo diaDeLaSemanaSegun
  Definir num Como Entero
  Escribir 'Ingresa un numero correspondiente al dia de la semana: '
  Leer num
  Segun num Hacer
    1:
      Escribir 'Domingo'
    2:
      Escribir 'Lunes'
    3:
      Escribir 'Martes'
    4:
      Escribir 'Miercoles'
    5:
      Escribir 'Jueves'
    6:
      Escribir 'Viernes'
    7:
      Escribir 'Sabado'
  De Otro Modo:
    Escribir 'Error. Debes ingresar un numero entre 1 y 7'
  Fin Segun
FinAlgoritmo
```

