

**Realizado por:**  
**Adrián Sánchez Chacón**  
**Sergio Lechuga Márquez**  
**Javier Rodríguez López**

**Algoritmo** calculadora



# ÍNDICE

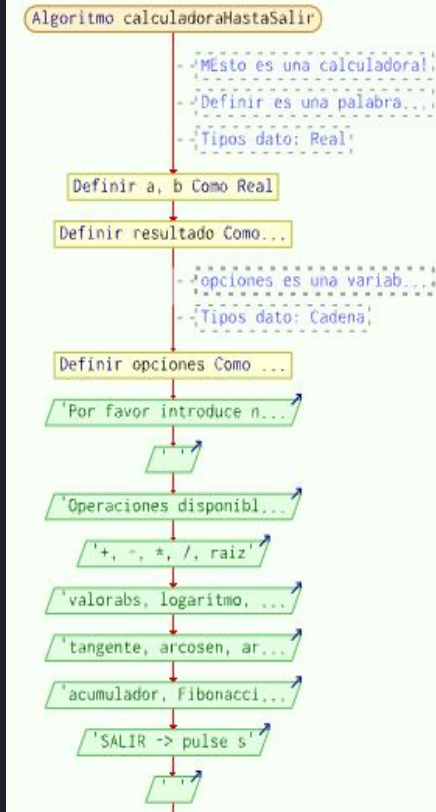
1. Estructura general
2. Estructura interna
3. Funcionalidad

# Definir, como real y como cadena.

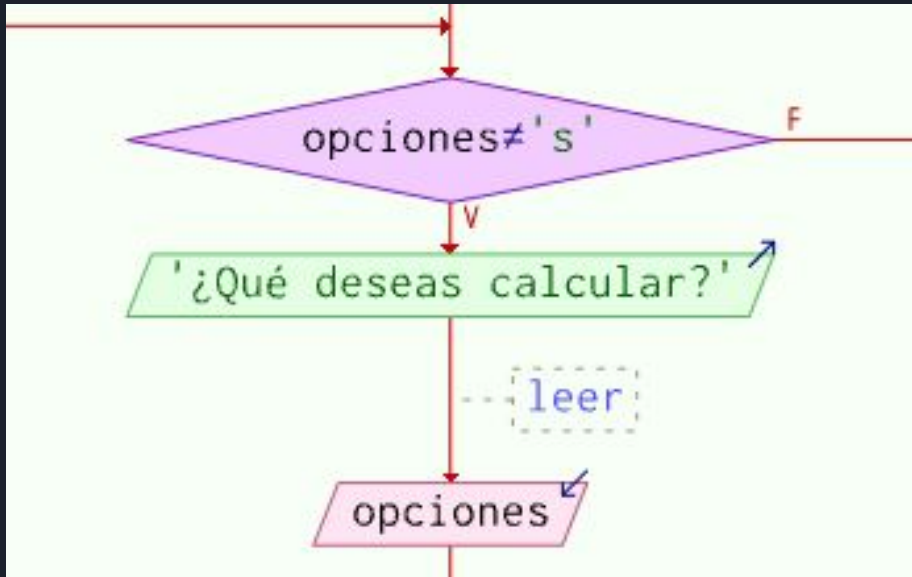
## Tipos de Datos Simples

Existen tres tipos de datos básicos:

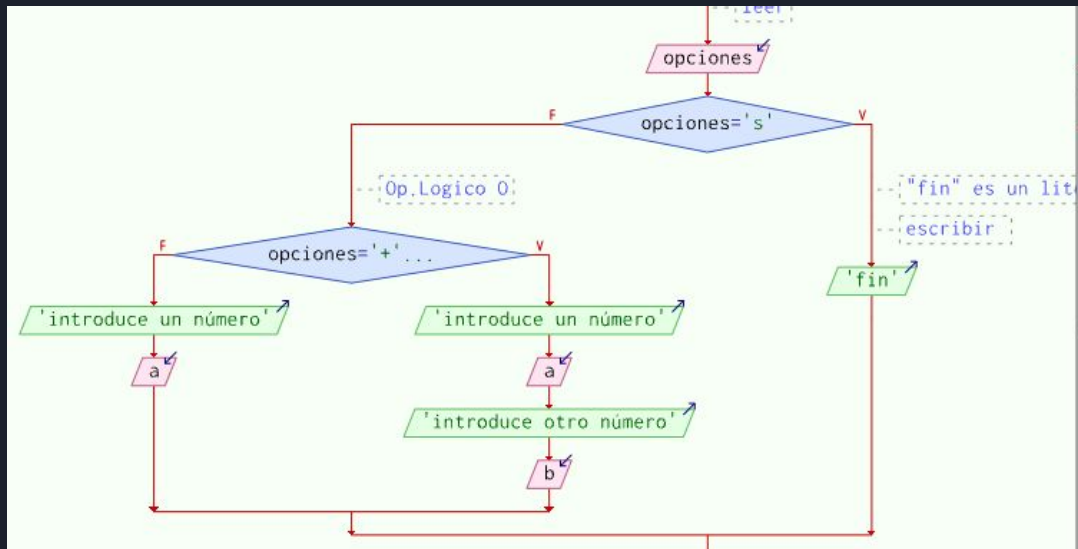
- ◆ **Númérico:** números, tanto enteros como reales. Para separar decimales se utiliza el punto. Ejemplos: 12 23 0 -2.3 3.14
- ◆ **Lógico:** solo puede tomar dos valores: VERDADERO o FALSO.
- ◆ **Carácter:** caracteres o cadenas de caracteres encerrados entre comillas (pueden ser dobles o simples). Ejemplos 'hola' "hola mundo" '123' 'FALSO' 'etc'



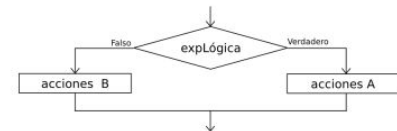
Aquí tendríamos un mientras,  
y un lee las opciones.



# Aqui tendríamos 2 si-entonces .



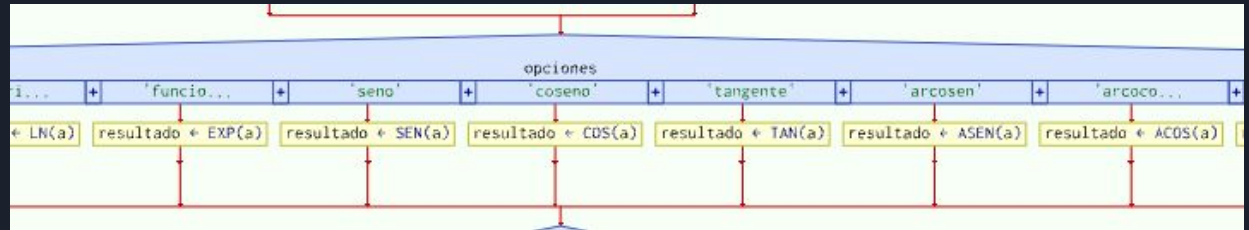
## Condicional Si-Entonces



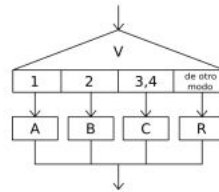
La secuencia de instrucciones ejecutadas por la instrucción Si-Entonces-SiNo depende del valor de una condición lógica.

Si <condición> Entonces  
<instrucciones>  
SiNo  
<instrucciones>  
FinSi

# Según



## Selección Múltiple



La secuencia de instrucciones ejecutada por una instrucción *Segun* depende del valor de una variable numérica.

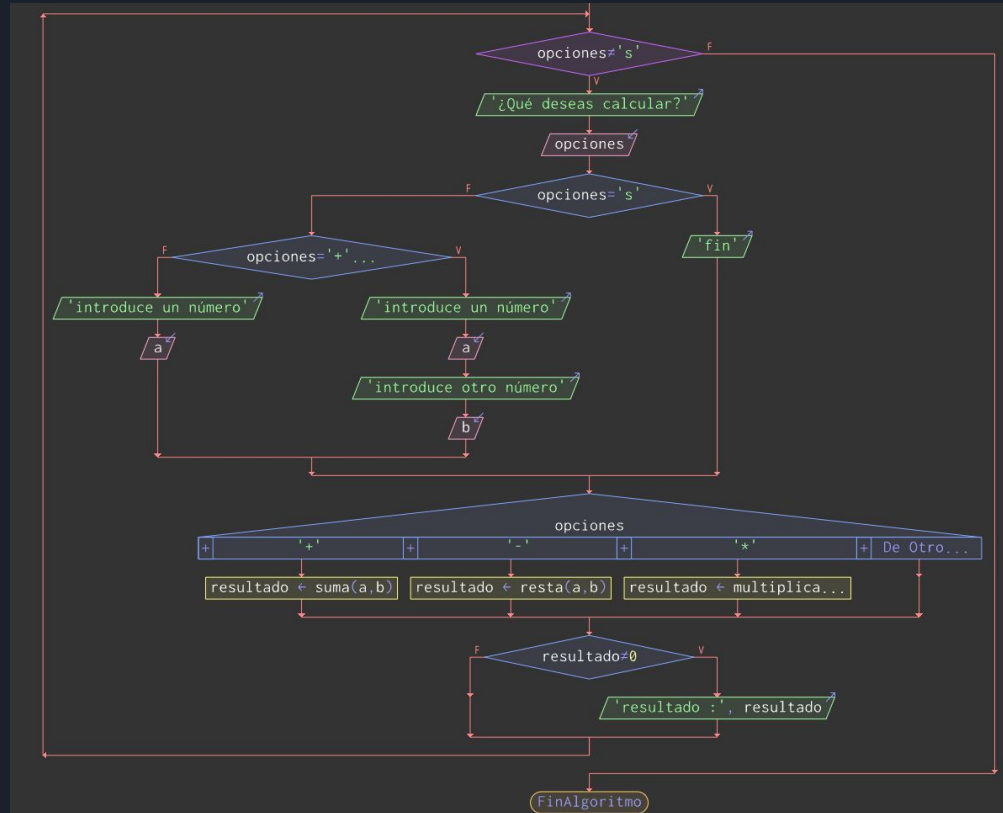
```
Segun <variable> Hacer
  <número1>: <instrucciones>
  <número2>,<número3>: <instrucciones>
  <...>
  De Otro Modo: <instrucciones>
FinSegun
```

# Si-entonces



# Estructura interna del Mientras (“While”) I

- Mientras:
  - Expresión Lógica
  - Se leen las Opciones
  - Si (Evaluar Opciones)
  - Según (Elegir Opciones)
  - 2º Si (Establecer Resultado)
  - FinMientras





## Estructura II → Evaluación de opciones y casos

```
//Programación indicativa -> iteración -> Mientras
// Mientras
Mientras opciones ≠ "s"
    Escribir "¿Qué deseas calcular?"
    //leer
    leer opciones
    si opciones = "s"
        // "fin" es un literal
        //escribir
        escribir "fin"
    sino
        // Op.Logico 0
        si opciones="+" o opciones="-" o opciones="*" o opciones="/" o opciones="Mayor" entonces
            Escribir "introduce un número"
            leer a
            Escribir "introduce otro número"
            leer b
        sino
            Escribir "introduce un número"
            leer a
        FinSi
    FinSi
segun opciones hacer
    caso "+":
        //Op.Relacionales -> "="
        resultado = suma(a,b)
    caso "-":
        resultado = resta(a,b)
    caso "*":
```

```
70 caso "Redondeo":
71     resultado = REDON(a)
72
73 caso "acumulador":
74     resultado = acumulador(a)
75
76 caso "Fibonacci":
77     hastaFibonacci(a)
78
79 caso "Mayor":
80     resultado2=esMayor(a,b)
81
82 De Otro Modo:
83     resultado = 0
84
85 FinSegun
86
87
```

## Estructura III → Funciones → acumulador(e)

```
115
116 Funcion resultado=acumulador(a)
117     //Contador -> para utiliza el contador i hasta i =10
118     //Acumulador -> "sumador" ejerce de acumulador
119     //Para
120     Para i = 0 hasta 10 Con Paso 0.87 Hacer
121         //Euler <- Constante
122         sumador = sumador + (Euler*a)
123     FinPara
124     resultado = sumador
125 FinFuncion
```

## Estructura IV → Procedimientos → hasta Fibonacci

```
127 Funcion hastaFibonacci(a)
128     elPrimero = 0
129     elSegundo = 1
130     Escribir "Los número de las serie Fibonacci hasta ", a, " son: "
131
132     Mientras elPrimero < a Hacer
133         Escribir elPrimero
134         siguiente = elPrimero + elSegundo
135         elPrimero = elSegundo
136         elSegundo = siguiente
137         //cambio de estado arriba
138
```

# ¿CÓMO FUNCIONA?

```
*** Ejecución Iniciada. ***  
Por favor introduce números positivos, que no sean 0  
  
Operaciones disponibles:  
+, -, *, /, raiz  
valorabs, logaritmo, funcionexp, seno, coseno  
tangente, arcosen, arcotang, redondeo  
acumulador, Fibonacci, Mayor  
SALIR -> pulse s  
  
¿Qué deseas calcular?  
>
```





FIN