

Pseudocódigo

Es un lenguaje parecido al natural, con cierta sintaxis y semántica que emula la programación de computadoras, en una forma más simple y comprensible. La idea de este lenguaje es poder escribir un algoritmo, diseñar la lógica y después poder reescribirlo en cualquier lenguaje de programación.

Estructuras de control

Asignación de variables

- asigne a x el valor de y
- $x \leftarrow y$
- $x := y$
- $x = y$

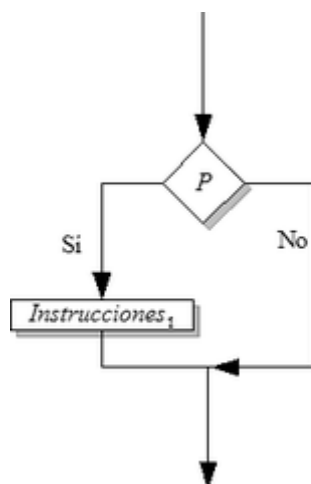
Secuencial

Son una serie de instrucciones que se ejecutan con el orden de aparición.

```
Instrucción1  
Instrucción2  
Instrucción3  
⋮  
Instrucciónn
```

Selectiva

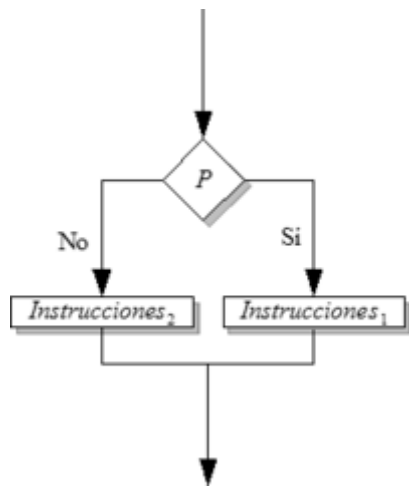
Esta estructura permite ejecutar una instrucción dependiendo de una anterior validación.



```
si P entonces  
  Instrucciones1  
fin si
```

Selectiva doble (alternativa)

Se puede elegir entre dos instrucciones dependiendo de una validación, si es verdadera se ejecuta una y sino otra.



```
si P entonces  
  Instrucciones1  
si no  
  Instrucciones2  
fin si
```

Selectiva múltiple

Podemos realizar también una serie de anidamientos de condiciones, por ejemplo si queremos evaluar depende el operador que operación matemática realizamos entre dos variables.

```
si Condición1 entonces  
  Instrucciones1  
si no si Condición2 entonces  
  Instrucciones2  
  ...  
si no si Condiciónn entonces  
  Instruccionesn - 1  
si no  
  Instruccionesn  
fin si
```

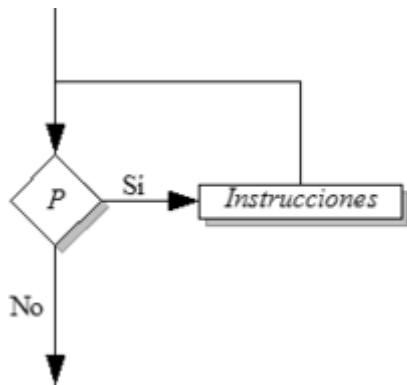
Selectiva múltiple-Casos

Equivalente a la estructura anterior.

```
seleccionar Indicador  
caso  $Valor_1$ :  
  Instrucciones_1  
caso  $Valor_2$ :  
  Instrucciones_2  
  ⋮  
  ⋮  
caso  $Valor_{n-1}$ :  
  Instrucciones_{n-1}  
[en otro caso:  
  Instrucciones_n]  
Fin seleccionar Indicador
```

Iterativa

Permiten repetir una serie de pasos finitos una instrucción, siempre y cuando se cumpla la condición.



```
mientras P hacer  
  Instrucciones  
fin mientras
```

O

```
hacer  
  Instrucciones  
mientras P
```

O

```
Instrucciones  
mientras P hacer  
  Instrucciones  
fin mientras
```

En la estructura siguiente se repite en base a un contador, cuando llega a su valor máximo definido, se corta su ciclo.

```
para  $i \leftarrow x$  hasta  $n$  hacer  
  Instrucciones  
fin para
```

Equivale en la estructura anterior a:

```
 $i \leftarrow x$   
mientras  $i \leq n$  hacer  
  Instrucciones  
   $i \leftarrow i + 1$   
fin mientras
```

Ejemplos

Antes de ver los ejemplos definamos que son las funciones y procedimientos: son subprogramas que sirven para realizar cierta acción. Los dos pueden recibir tantos parámetros como queramos, pero solo uno de los dos nos devuelve un valor, las funciones.

Imaginemos como funciona a $f(x) = x * 2$, le damos un valor a x y nos devuelve ese valor multiplicado por 2.

Método de ordenamiento por burbujeo

```
procedimiento Ordenar( $L$ )  
  //  $L = (L_1, L_2, \dots, L_n)$  es una lista con  $n$  elementos//  
  hacer  
    intercambio  $\leftarrow$  falso  
    para  $i \leftarrow 1$  hasta  $n - 1$  hacer  
      si  $L_i > L_{i+1}$  entonces  
        intercambiar( $L_i, L_{i+1}$ )  
        intercambio  $\leftarrow$  verdadero  
      fin si  
    fin para  
    mientras intercambio = verdadero  
  fin procedimiento
```

Función para calcular a^n (un número a elevado a potencia n).

```
función potencia( $a, n$ )  
//Este algoritmo calcula  $a^n$  (con  $a$  y  $n$  numeros naturales)  
 $i \leftarrow n$   
 $r \leftarrow 1$   
 $x \leftarrow a$   
mientras  $i > 0$  hacer  
  si  $i$  es impar entonces  
     $r \leftarrow r \times x$   
  fin si  
   $x \leftarrow x \times x$   
   $i \leftarrow i \div 2$   
fin mientras  
devolver  $r$   
fin función
```