



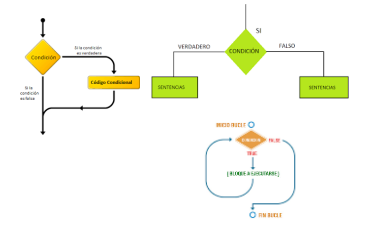
# Conceptos de Algoritmos Datos y Programas

# CADP – TEMAS

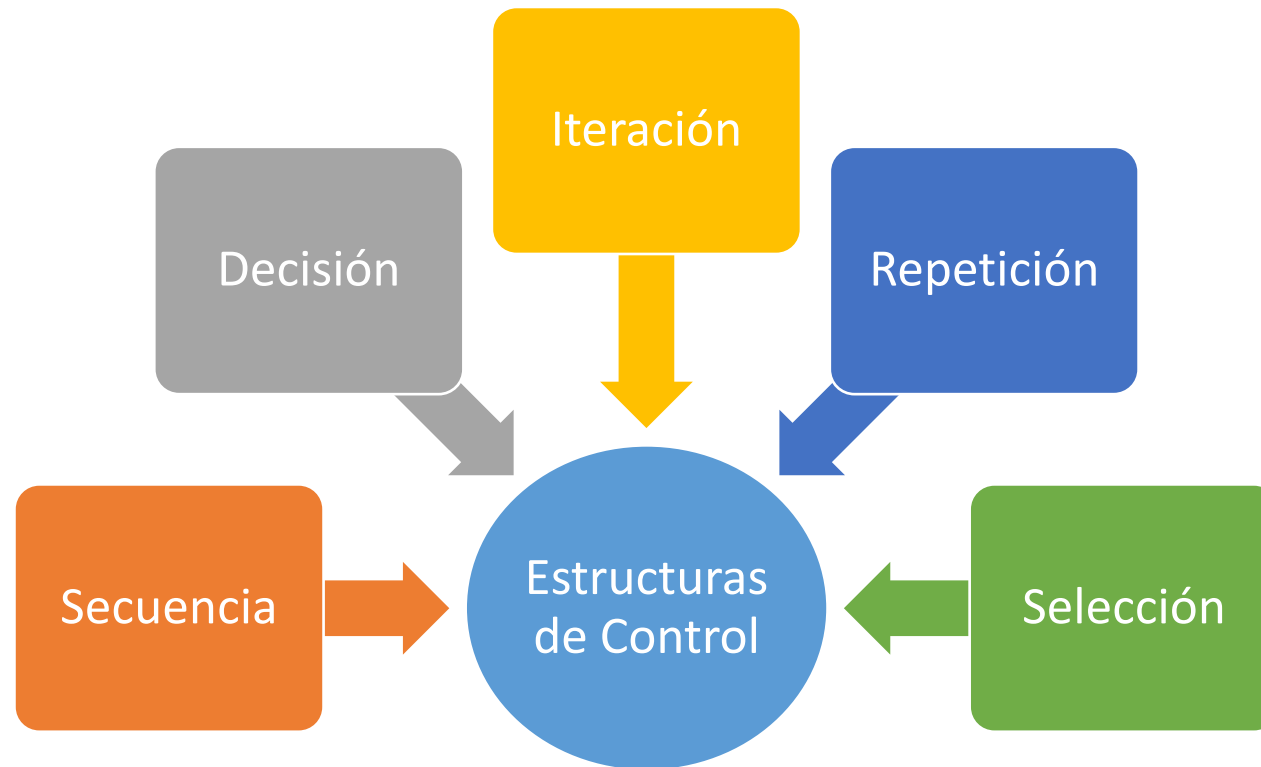


- Estructura de control
- Estructuras de control repetitivas
- Estructura de control FOR

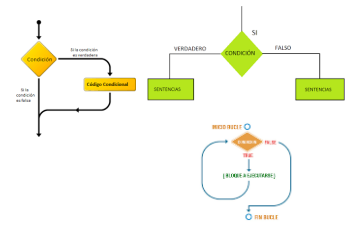
# CADP – ESTRUCTURAS DE CONTROL



Todos los lenguajes de programación tienen un conjunto mínimo de instrucciones que permiten especificar el control del algoritmo que se quiere implementar. Como mínimo deben contener: secuencia, decisión e iteración.



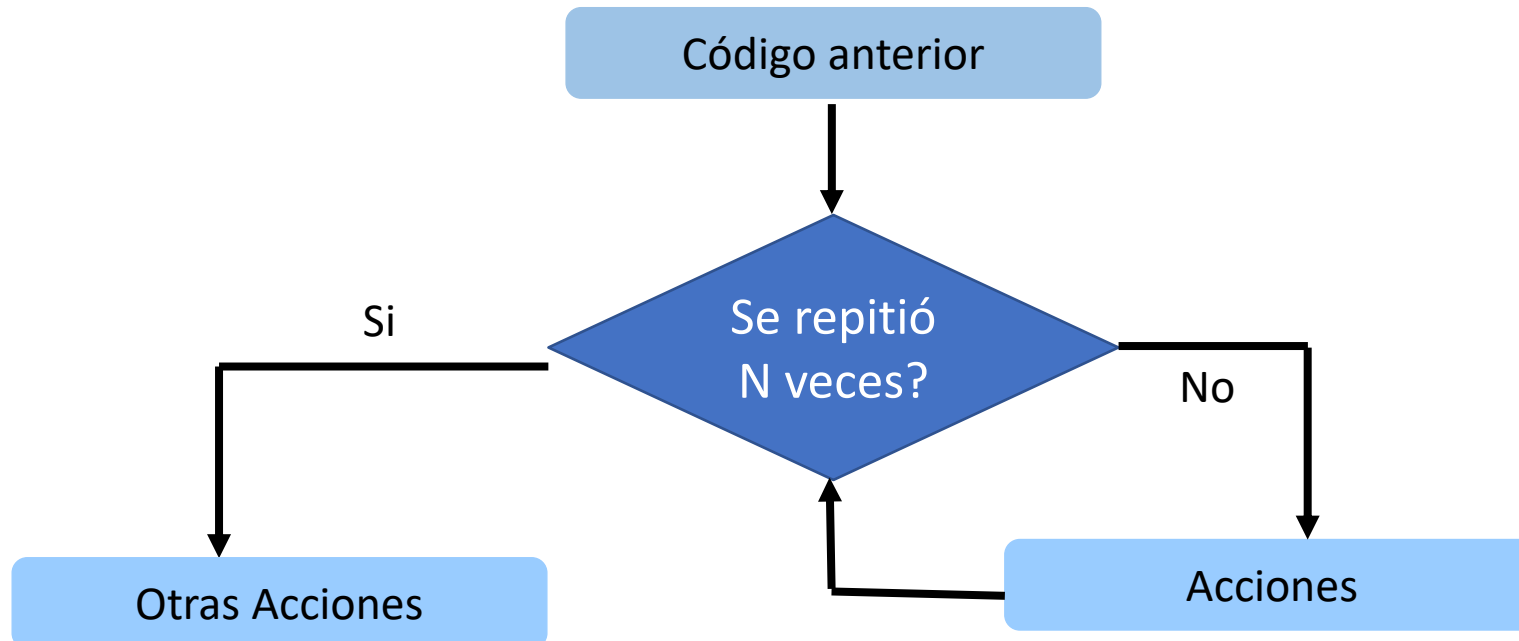
# CADP – ESTRUCTURAS DE CONTROL



## REPETICION

Es una extensión natural de la secuencia. Consiste en repetir N veces un bloque de acciones.

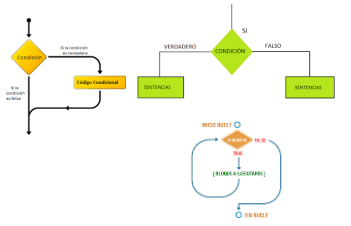
Este número de veces que se deben ejecutar las acciones es fijo y conocido de antemano



A qué estructura de control vista en el entorno del robot se parece?.

*Cómo es la sintaxis?*

# CADP – ESTRUCTURAS DE CONTROL REPETICION



```
for indice := valor_inicial to valor_final do  
    accion 1;
```



más de una  
acción

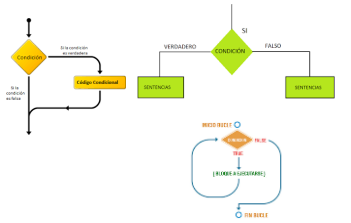
```
for indice := valor_inicial to valor_final do  
    begin  
        accion 1;  
        accion 2;  
    end;
```

Qué es el  
índice?

Dónde se  
declara?

Qué son valor  
final e inicial?

# CADP – ESTRUCTURAS DE CONTROL REPETICION



Ejemplo 0:

```
For i := 1 to 10 do  
    accion;
```

¿De qué tipo es el índice i?  
¿qué valores toma i?

Ejemplo 1:

```
For i := 'A' to 'H' do  
    accion;
```

¿De qué tipo es el índice i?  
¿qué valores toma i?

Ejemplo 2:

```
For i:= False to True do  
    accion;
```

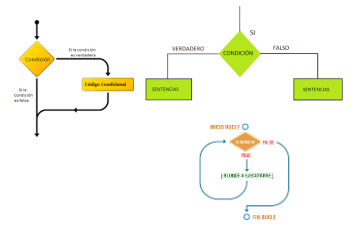
¿De qué tipo es el índice i?  
¿qué valores toma i?

Ejemplo 3:

```
For i := 20 to 18 do  
    accion;
```

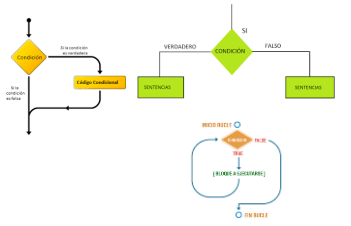
```
For índice := 20 downto 18 do  
    begin  
        accion;  
        accion;  
    end;
```

# CADP – ESTRUCTURAS DE CONTROL REPETICION



- La variable índice debe ser de tipo ordinal
- La variable índice no puede modificarse dentro del lazo
- La variable índice se incrementa y decrementa automáticamente
- Cuando el for termina la variable índice no tiene valor definido.

# CADP – ESTRUCTURAS DE CONTROL REPETICION



Realizar un programa que lea precios de 10 productos que vende un almacén. Al finalizar informe la suma de todos los precios leídos.

- Qué valor es el precio?
- Cuál es la condición de fin?
- Cómo calculo la suma

100,5

56,5

15

10

12,5

14

7,5

150,00

25,40

78,50

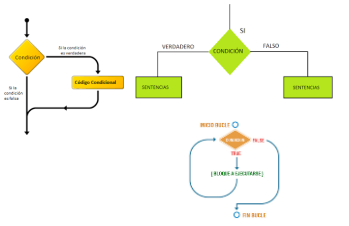


informa

**474,4**



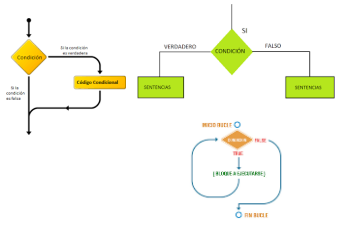
# CADP – ESTRUCTURAS DE CONTROL REPETICION



```
Program uno;  
var  
    precio,total:real;  
    i:integer;  
begin  
    total := 0;  
    for i:= 1 to 10 do  
        begin  
            read (precio);  
            total:= total + precio;  
        end;  
    write ("La suma de los precios de los  
        productos del almacén son: ",total);  
end.
```

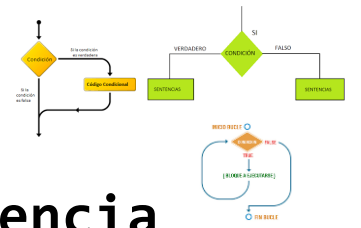
Qué modificaría si  
quiere informar al final,  
también el precio del  
5to producto?

# CADP – ESTRUCTURAS DE CONTROL REPETICION



```
Program uno;
var
    quinto,precio,total:real;
    i:integer;
begin
    total := 0;
    for i:= 1 to 10 do
        begin
            read (precio);
            if (i=5) then
                quinto:= precio;
                total:= total + precio;
            end;
        write ("La suma de los precios de los
                productos del almacén son: ",total);
        write ("El precio del quinto producto es: ",quinto);
    end.
```

# CADP – ESTRUCTURAS DE CONTROL



Qué crees que imprime el programa, si se leyera esta secuencia de números:

```
Program uno;
var
    i,num1,num2:integer;
Begin
    num2:= 0;
    for i:= 1 to 5 do
        begin
            read (num1);
            while (num1 mod 2 = 0) do
                begin
                    num2:= num2 + 1;
                    read (num1);
                end;
            end;
            write (num2);
        end;
    end.
```

4  
8  
126  
5  
3  
6  
1  
1568  
6  
10  
7  
19  
22  
24  
3

# CADP – ESTRUCTURAS DE CONTROL



Qué crees que imprime el programa, si se leyera esta secuencia de números:

```
Program uno;  
var  
    i,j,num1,num2:integer;  
Begin  
    num2:= 0;  
    for i:= 1 to 3 do  
        begin  
            read (num1);  
            for j:= 1 to 2 do  
                begin  
                    if (num1 mod 2 = 1) then  
                        num2:= num2+1;  
                    read (num1);  
                end;  
                read (num1);  
            end;  
            write (num2);  
        end;  
    end.
```

4  
7  
126  
5  
3  
6  
1  
1568  
6  
10  
7  
19  
22  
24  
3