

COMPUTACIÓN I

CLASE DE TEORÍA 5

(CI-2125, CI-2127)

Profesor

Darwin Rocha
(DarwinRocha@usb.ve)

CICLOS

REPASO CLASE 4

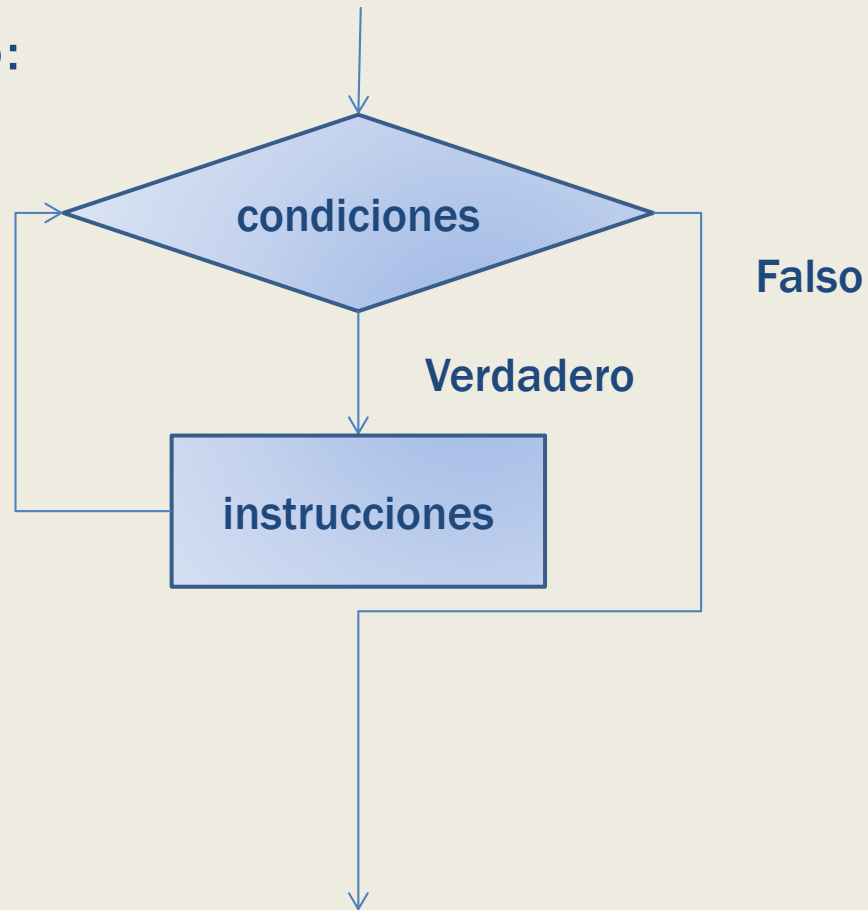
Hay tres tipos de ciclos o bucles.

- **Mientras** (se cumpla la condición)
- **Repetir-hasta** (que se cumpla la condición)
- **Para** (una condición inicial hasta una condición final)

CICLO: MIENTRAS

REPASO CLASE 4

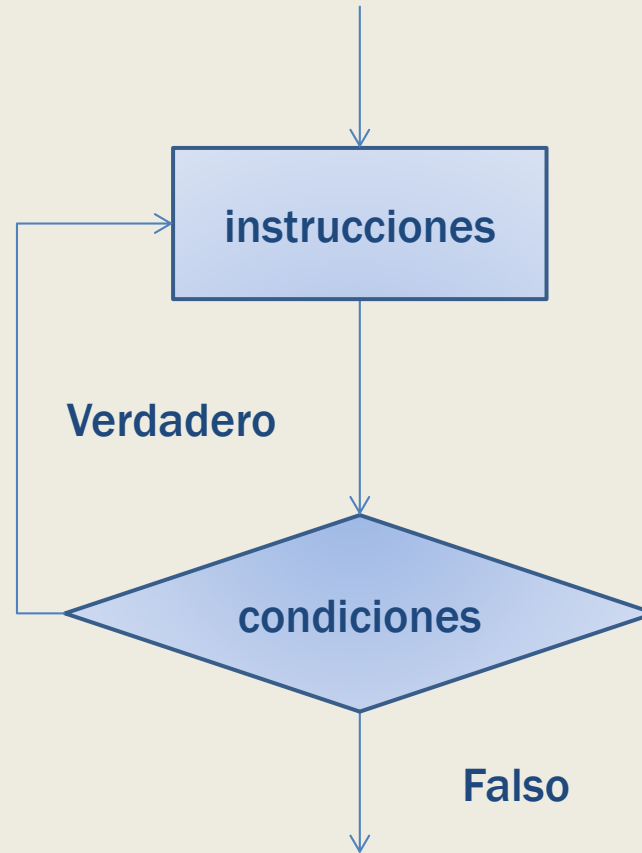
Diagrama de flujo:



CICLO: REPETIR-HASTA

REPASO CLASE 4

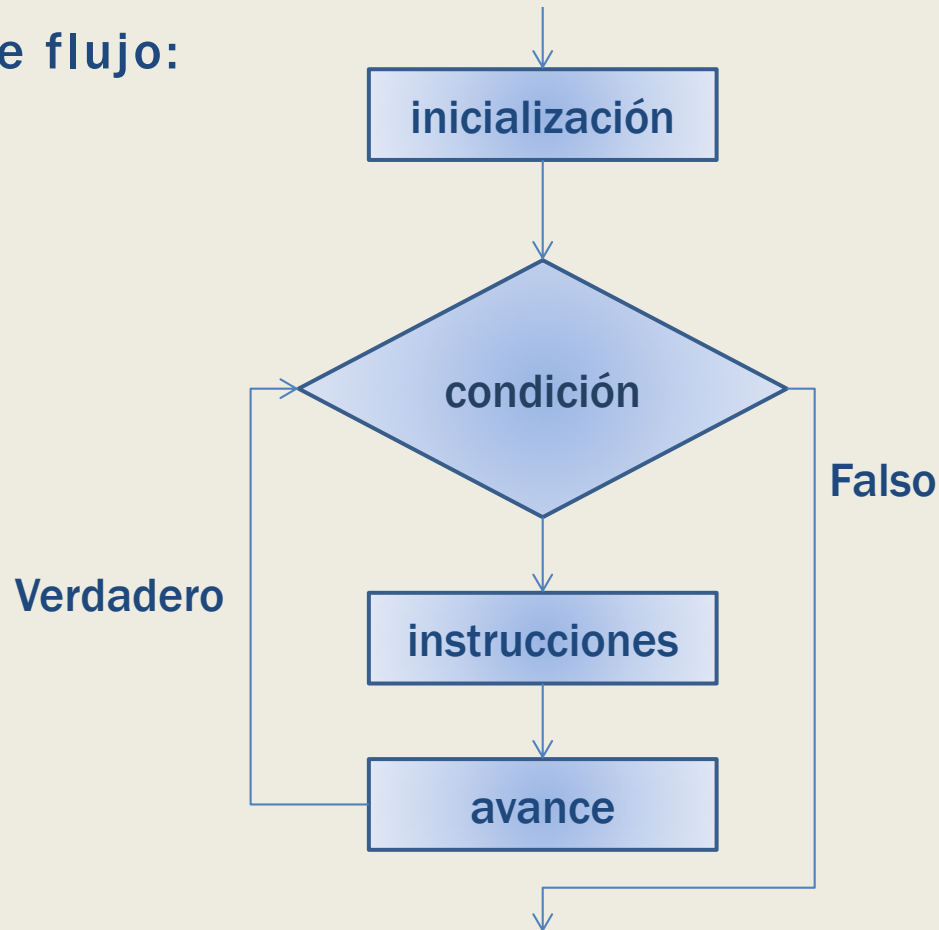
Diagrama de flujo:



CICLO: PARA

REPASO CLASE 4

Diagrama de flujo:



SUB-PROGRAMAS

Los sub-programas permiten desarrollar soluciones de problemas complejos, a través del método descendente (top-down)

SUB-PROGRAMAS

La idea es separar un problema grande o muy complejo en subprogramas mas sencillos de elaborar y de entender.

FUNCIONES Y PROCEDIMIENTOS

Existen dos tipo de subprogramas las funciones y los procedimientos.

- **funciones**: son subprogramas que retornan **un** resultado.
- **Procedimientos**: son subprogramas que **no** retornan resultado.

FUNCION: ES_PAR(NUM)

EJEMPLO 5.A

Escriba en diagrama de flujo, un subprograma que reciba un numero entero y devuelva un valor booleano: *true* si es par y *false* en caso contrario.

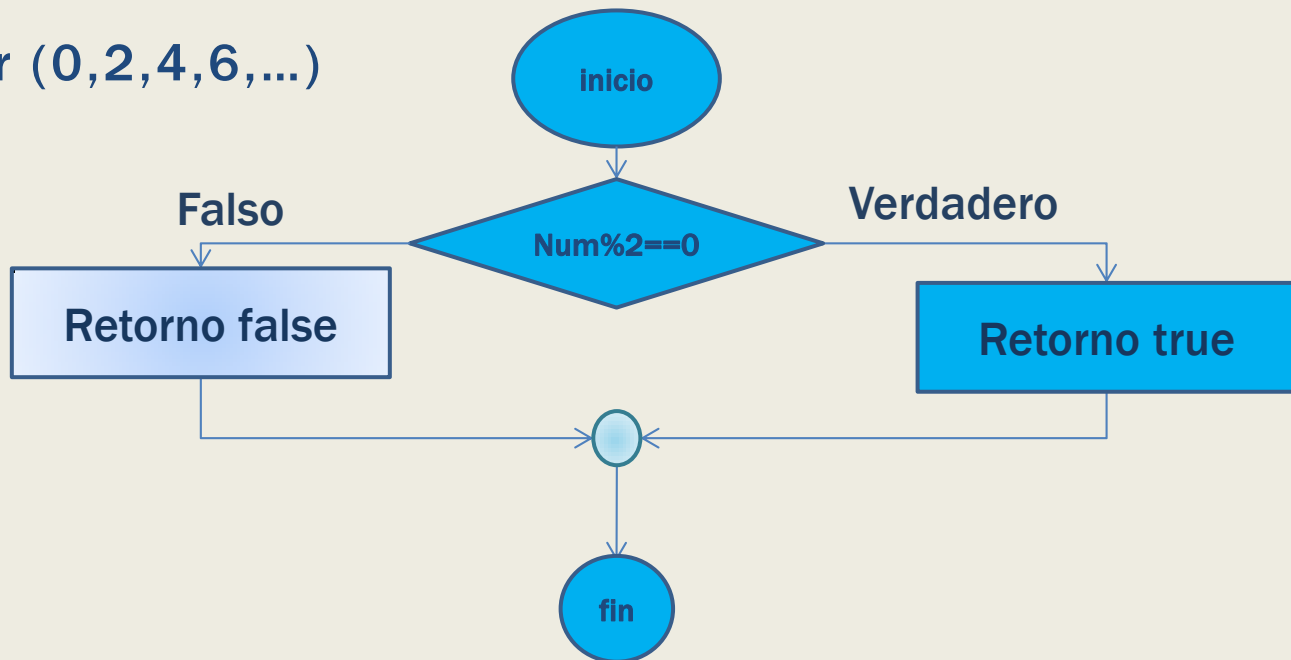
FUNCION: ES_PAR(NUM)

EJEMPLO 5.A

Declaracion:

Booleano = esPar(num)

Si num es par (0,2,4,6,...)



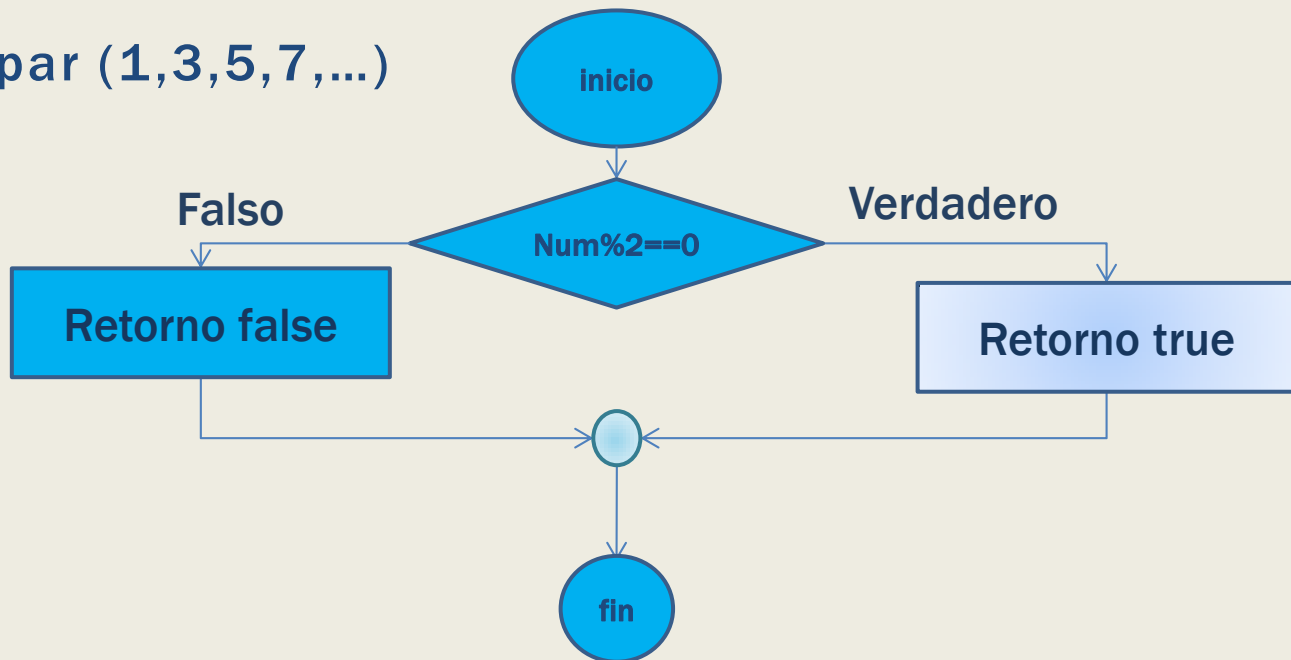
FUNCION: ES_PAR(NUM)

EJEMPLO 5.A

Declaracion:

Booleano = esPar(num)

Si num es impar (1,3,5,7,...)



PROCEDIMIENTOS

5.B

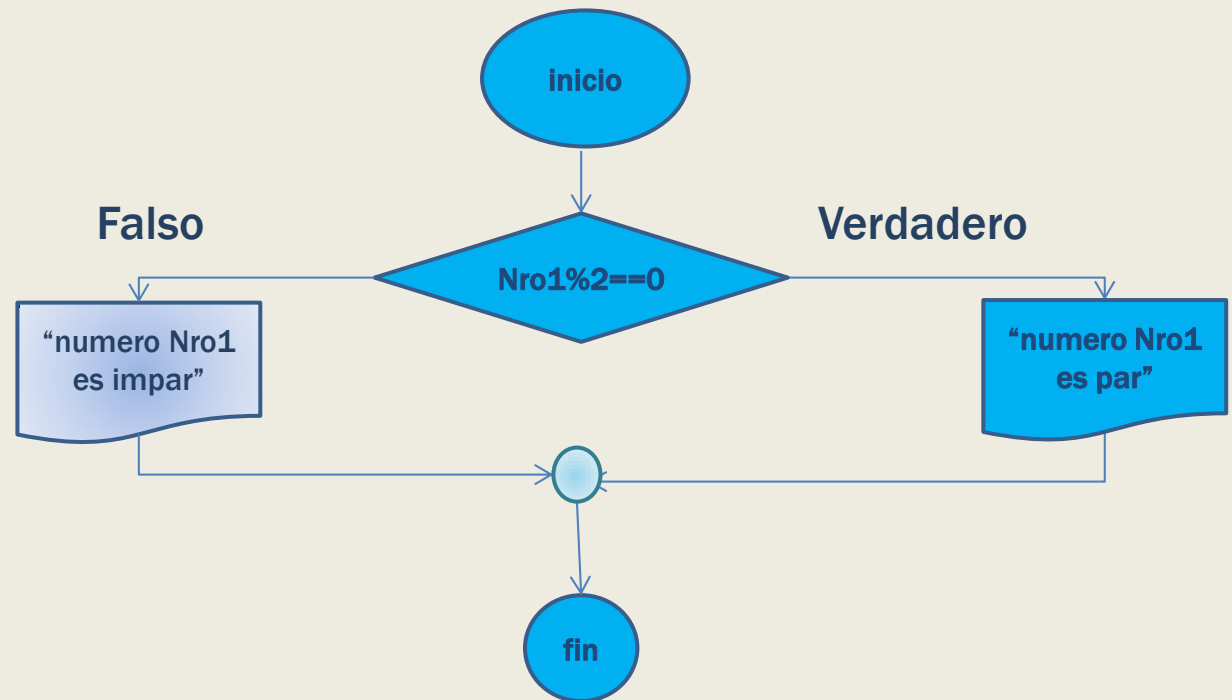
Escriba en diagrama de flujo, un subprograma que reciba un numero entero y escriba en pantalla si es par o impar.

EJEMPLO 5.B

Declaracion:

es_par(Nro1)

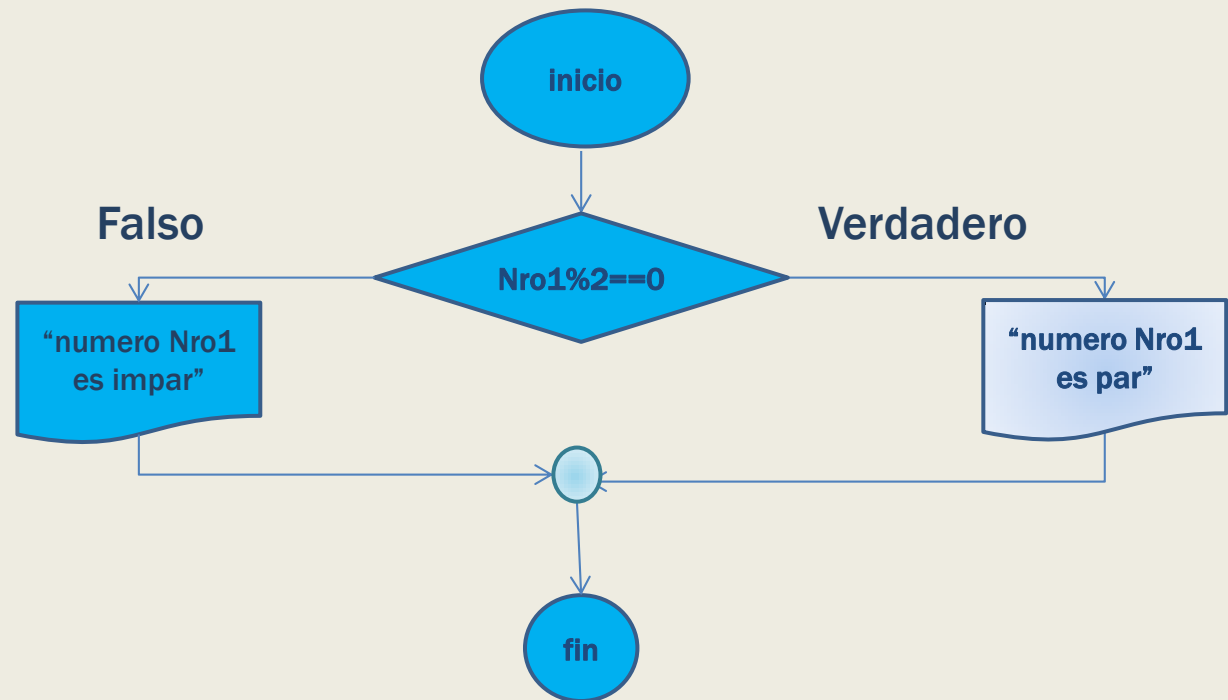
Si Nro1 es 34:



EJEMPLO 5.B

Declaracion:
es_par(Nro1)

Si Nro1 es 7 :



PASE DE PARÁMETROS

Los parámetros que recibe un subprograma pueden ser por valor o por referencia.

- ***Por valor***: el subprograma recibe una copia el valor que tiene la variable al momento de llamar al subprograma. *Los cambios hechos al valor de la variable en el subprograma no se reflejan en el programa que lo invoco.*
- ***Por referencia***: el subprograma recibe un apuntador a la variable y *este podrá consultar/modificar el valor de la variable.*

PASE DE PARAMETROS POR VALOR

Escriba un diagrama de flujo que reciba un entero N , y retorne su valor absoluto. *Nota:* el valor absoluto de un positivo es el mismo número, y el de un negativo es el número por (-1)

N1=ABSOLUTO(N2)

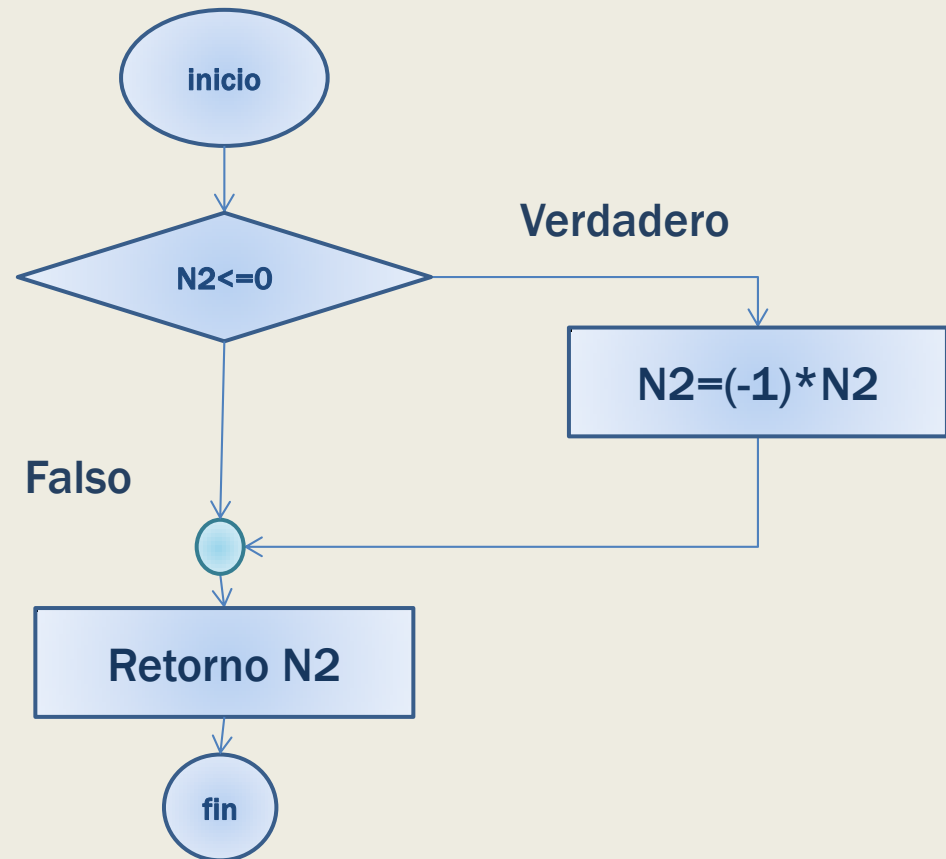
EJEMPLO 5.C

$N1 = \text{absoluto}(N2)$

Nótese que el valor de N2 cambia en el subprograma Pero no el programa principal

Si el subprograma recibe 7
Devuelve 7.

Si el subprograma recibe -7
Devuelve 7.



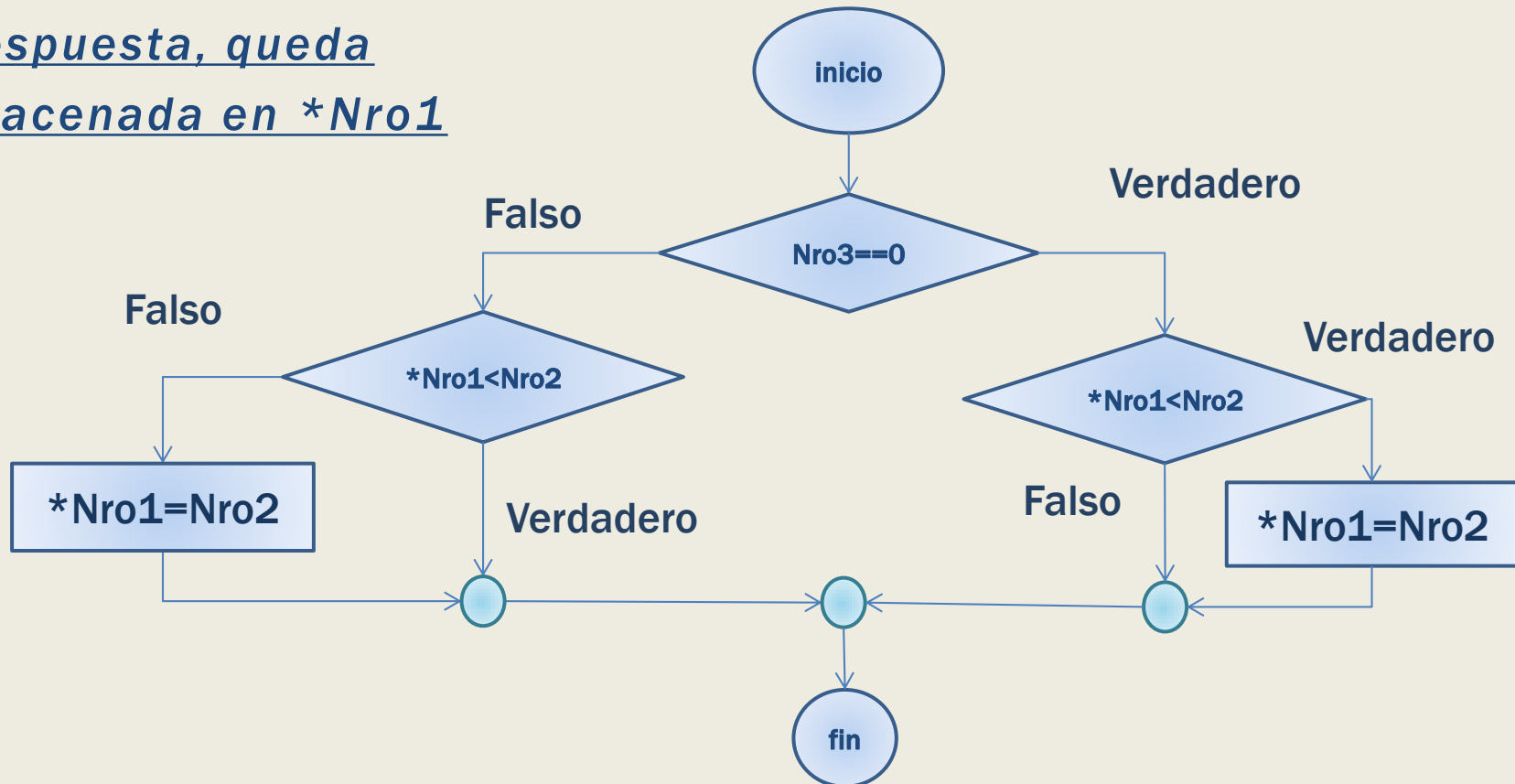
PASE DE PARAMETROS POR REFERENCIA

Escriba un diagrama de flujo que reciba, tres enteros Nro1, Nro2, Nro3, y devuelva el mayor entre los dos primeros parámetros si el tercero es 0, en caso de contrario devolver el menor de los dos primeros parámetros. El resultado debe ser devuelto por referencia en Nro1.

MAYOR(*NRO1, NRO2, NRO3)

EJEMPLO 5.D

Nótese que en cualquier caso,
la respuesta, queda
almacenada en *Nro1



COMO LLAMAR A UN SUBPROGRAMA

Para llamar a un subprograma basta con colocar el nombre del subprograma y entre paréntesis los parámetros que recibe, en el mismo orden que esperan ser recibidos en el subprograma.

COMO LLAMAR A UN SUBPROGRAMA

Especificando además con un & cuales son por referencia y asignando su resultado a una variable en caso de devolver algún valor.

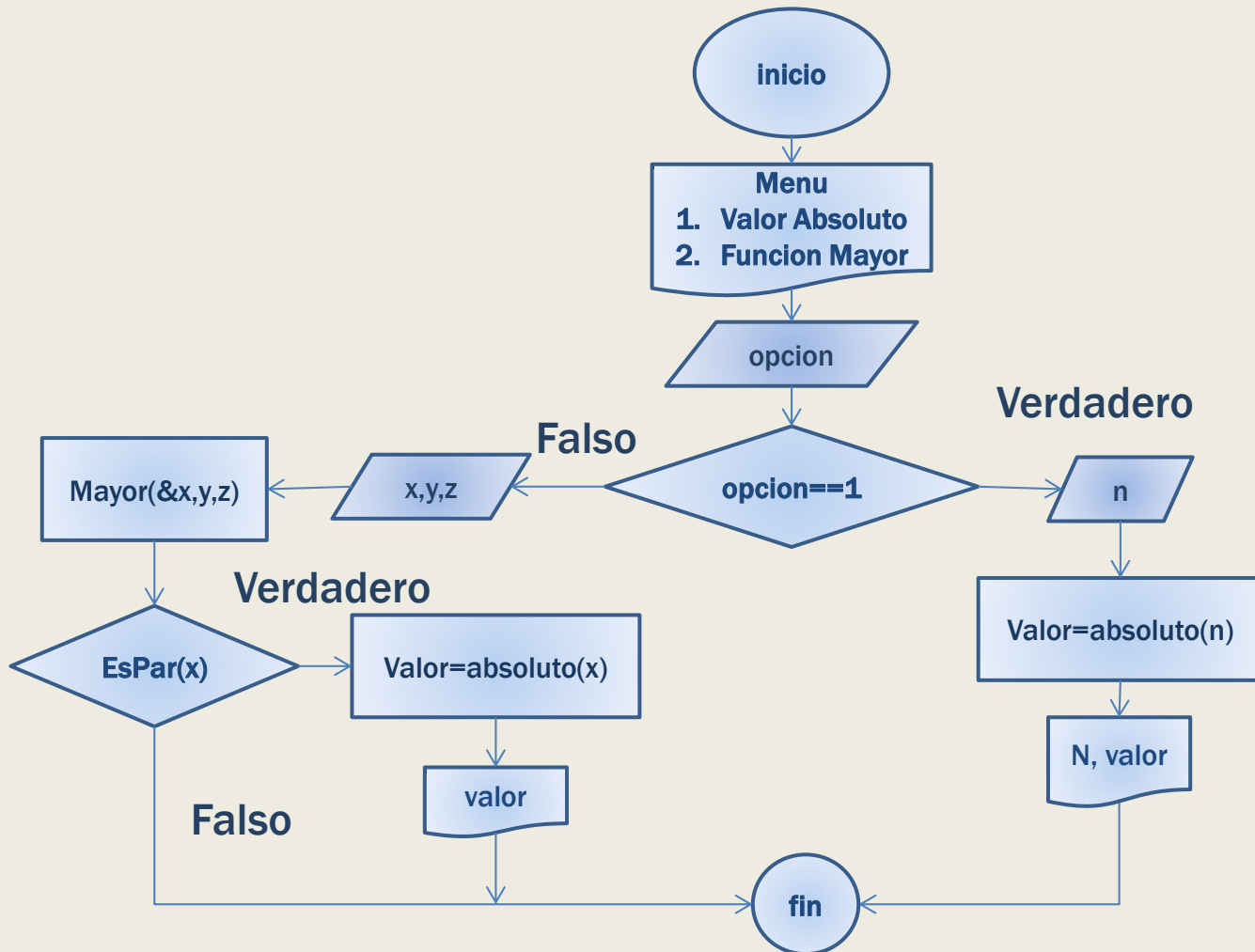
Nota: la variable debe ser del mismo tipo que espera ser recibido.

PROGRAMA PRINCIPAL

EJEMPLO 5.E

Se desea crear un programa principal que invoque los subprogramas creados previamente.

EJEMPLO 5.E



¿PREGUNTAS?

Cubículo: MYS-228A (previo acuerdo por correo)

Correo: darwinrocha@usb.ve

También pueden hacer consulta a través de los foros de moodle.

Romanos 16:3

“Pon todo lo que hagas en manos del Señor, y tus planes tendrán éxito”