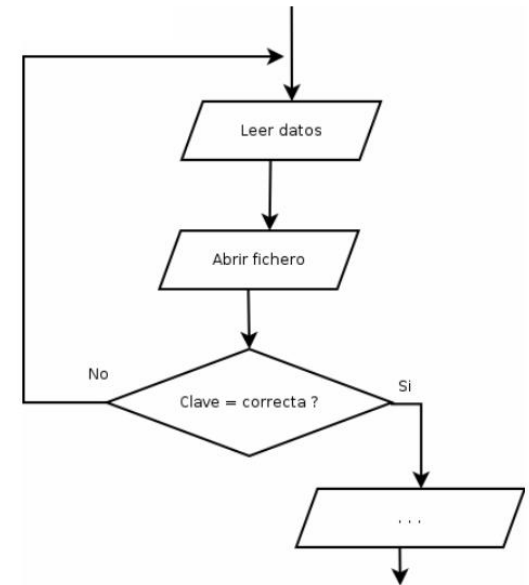
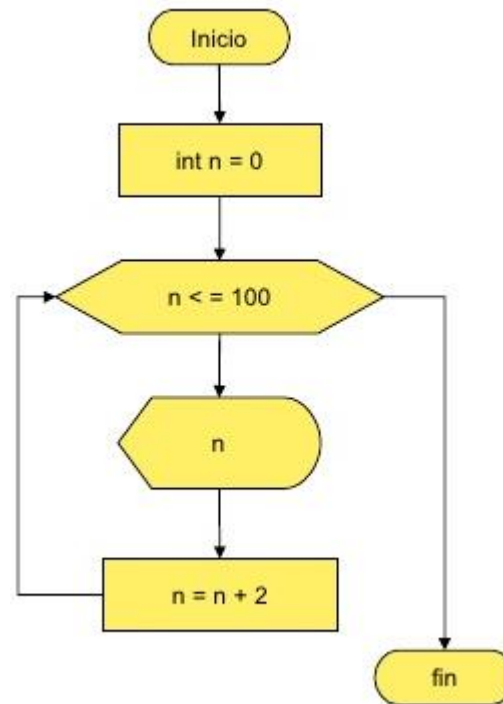
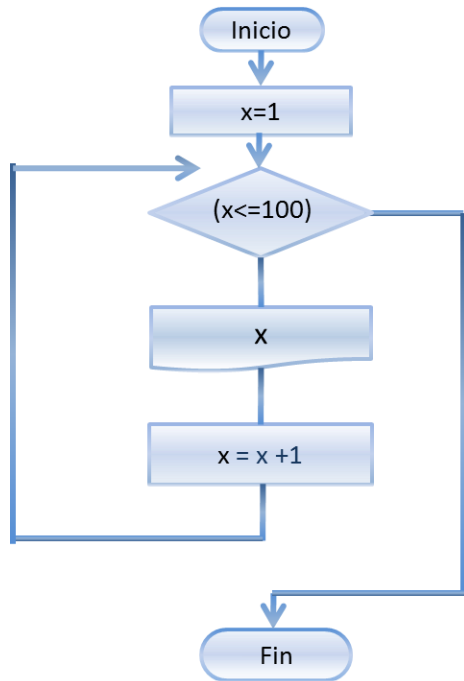
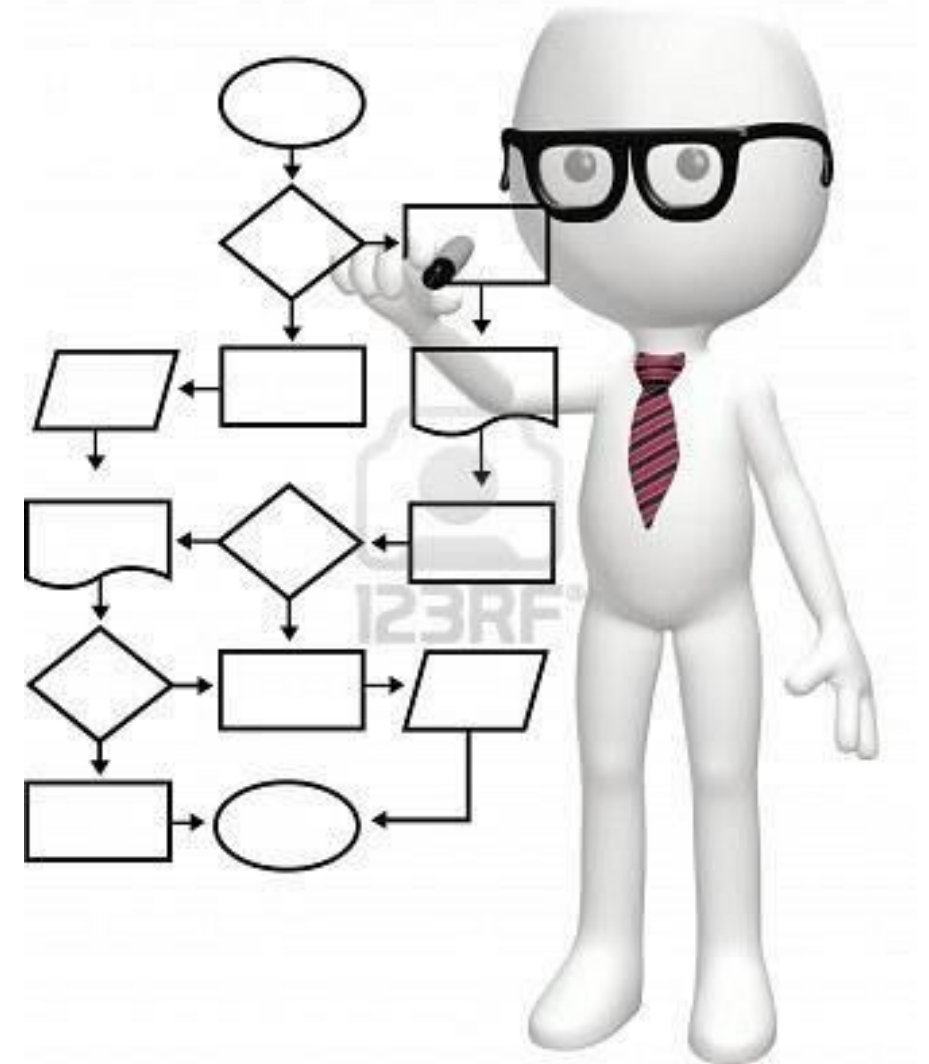


ESTRUCTURAS REPETITIVAS



TEMAS:

1. Contadores
2. Acumuladores
3. Banderas
4. Estructuras de repetición



Objetivo

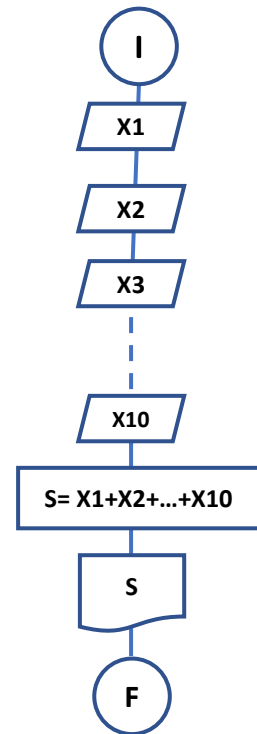
- Determinar la función de cada uno de los recursos (*contador, acumulador y bandera*) que se utilizan en una estructura repetitiva.
- Definir qué es una estructura repetitiva.
- Diferenciar las estructuras repetitivas según su comportamiento.
- Identificar la mejor estructura repetitiva según su eficiencia para un caso específico.

ESTRUCTURA DE REPETICIÓN O ITERACIÓN

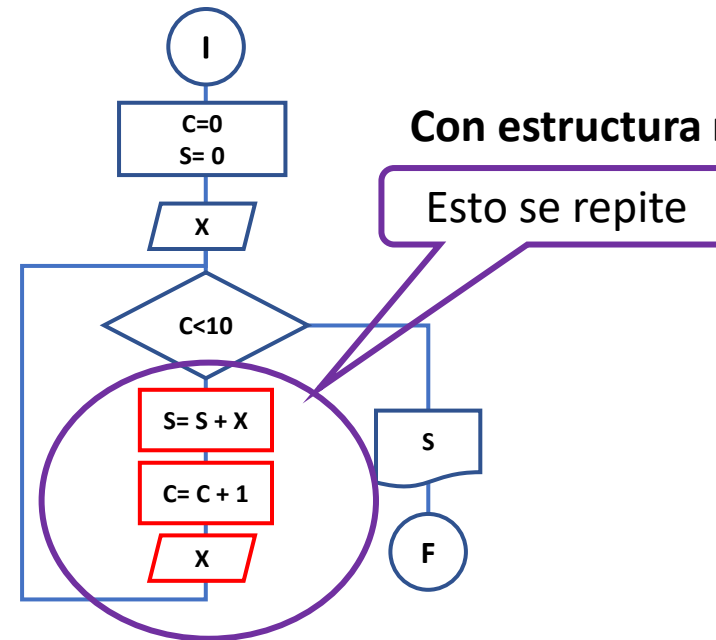
Las estructuras repetitivas se utilizan cuando se quiere que un conjunto de instrucciones se ejecuten ***un cierto número finito de veces***.

Por ej. Ingrese 10 valores y muestre la suma de los mismos.

Con estructura secuencial



Con estructura repetitivas



Contadores Acumuladores y Banderas

CONTADOR



Es una variable cuyo valor se *incrementa* o *decrementa* en una cantidad **constante** cada vez que se produce un determinado suceso o acción.

Su función: Contar sucesos o acciones internas de un bucle. Se debe **inicializar** al contador asignándole un valor inicial. Se situará antes y fuera del bucle. La variable contador es de **tipo entero**.

Sintaxis:

IDENTIFICADOR = IDENTIFICADOR + VALOR CONSTANTE

C=0

Por ejemplo: **C = C + 1**

ACUMULADOR



Es una variable cuyo valor se incrementa o decrementa en una **cantidad variable** en **cada iteración o ciclo** de un proceso repetitivo.

Su función: se utiliza con frecuencia para sumar una determinada cantidad de valores, por lo tanto la variable debe ser de tipo entero o real. También deber ser *inicializada* antes de ser usada.

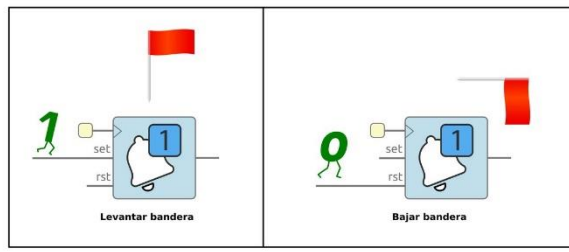
Sintaxis:

IDENTIFICADOR = IDENTIFICADOR + VALOR VARIABLE

S = 0

Por ejemplo: **S = S + X**

VARIABLE BANDERA



Una variable bandera es aquella que presenta estados, y según lo que ocurra cambia el estado.

Es utilizada dentro de la **condición de un bucle**, para determinar cuándo un bucle se sigue iterando o cuando no. De esta manera una variable bandera debe ser de tipo **booleano o entero**.

Por ejemplo:

IDENTIFICADOR = Falso (VALOR INICIAL)

Si ocurre un suceso: **IDENTIFICADOR = Verdadero o 1**

Si ocurre otro suceso: **IDENTIFICADOR = Falso o 0**

EJEMPLO DE CONTADOR, ACUMULADOR Y BANDERA

Supongamos que somos cajeros de un banco. El cajero puede descansar 1 hora por jornada cada vez que atendió a 50 clientes o que haya pagado más de \$100000. El descanso del cajero será controlado por su superior para ver si llegó a los 50 clientes, o si llegó a los 100000 y si descansó o no.

Entonces vamos a usar **c** para contar los clientes y **s** para contar el dinero pagado y **b** para indicar su descanso.

Sintaxis:

Iniciar las variables

$c=0 \rightarrow$ Arranca en 0

$s=0 \rightarrow$ Arranca en 0

$b=0 \rightarrow$ Sin descanso

Función de las variables

$c = c + 1 \rightarrow$ cuenta clientes

$s = s + \text{dinero} \rightarrow$ Acumula dinero

Algoritmo

Entra el cliente 1

Cajero paga al cliente X dinero

Controla el supervisor si $c = 50$ o si $s > 100000$

Si eso es verdad verifica si **no** ocupó su descanso $b=0$

si $b=0$ le corresponde el descanso y cambia a $b=1$

si b no es $=0$ no le corresponde el descanso (siga trabajando)

Se incrementa el contador $c=c+1$

Se actualiza el acumulador $s=s+ X \text{ dinero}$

Si $c < 50$ o si s no es mayor a 100000 pasa el siguiente cliente

Cuándo utilizaríamos un proceso repetitivo?

- Escribir algo en pantalla cierta cantidad de veces
- Hacer una operación matemática cierta cantidad de veces.

Entonces, a todo proceso que se repite cierto número de veces dentro de un pseudocódigo o un programa se les llama **bucle** o **ciclo**.

Y la expresión lógica que controla el flujo repetitivo se denomina **Corte de Control**.

Estas estructuras son las siguientes:

1. Estructura **FOR**
2. Estructura **WHILE**
3. Estructura **DO WHILE**

ESTRUCTURA WHILE (MIENTRAS)

Mientras (condición sea verdadera) **hacer**

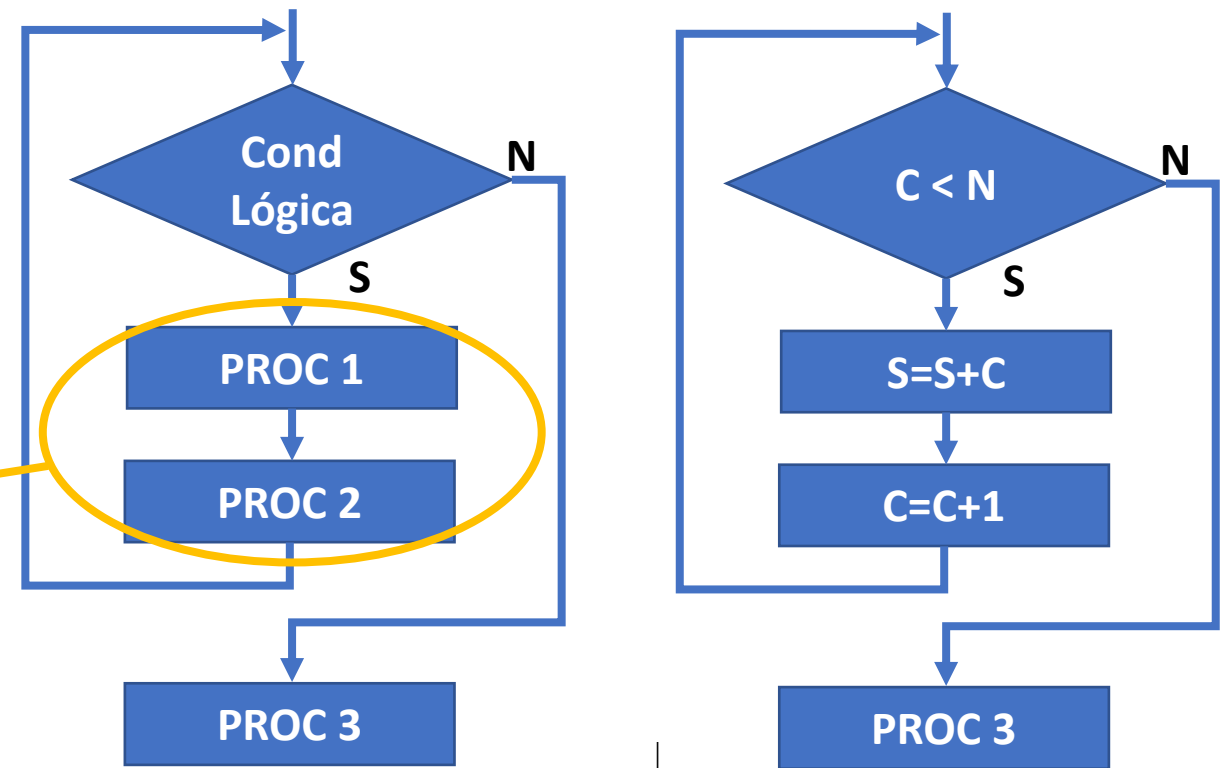
PROC 1

PROC 2

Fin_mientras

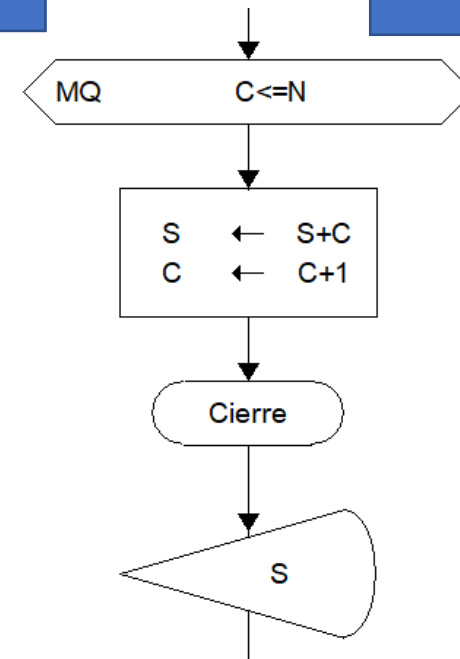
PROC 3

Proceso que
se repite



La expresión lógica se evalúa **antes** de la ejecución del bucle. Si la condición es verdadera se ejecuta las acciones incluidas en el bucle, caso contrario el control pasa a la sentencia siguiente al lazo.

Si la expresión lógica no se cumple cuando se ejecuta el bucle por primera vez, el conjunto de acciones del lazo **no se ejecuta nunca**.



ESTRUCTURA FOR (PARA)

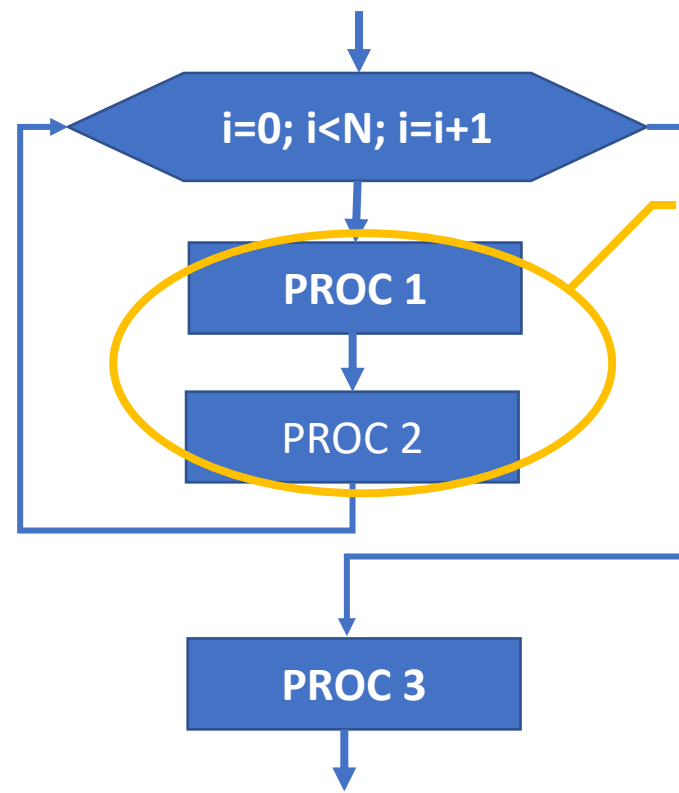
Para $i = 0$ hasta N de 1 en 1

PROC 1

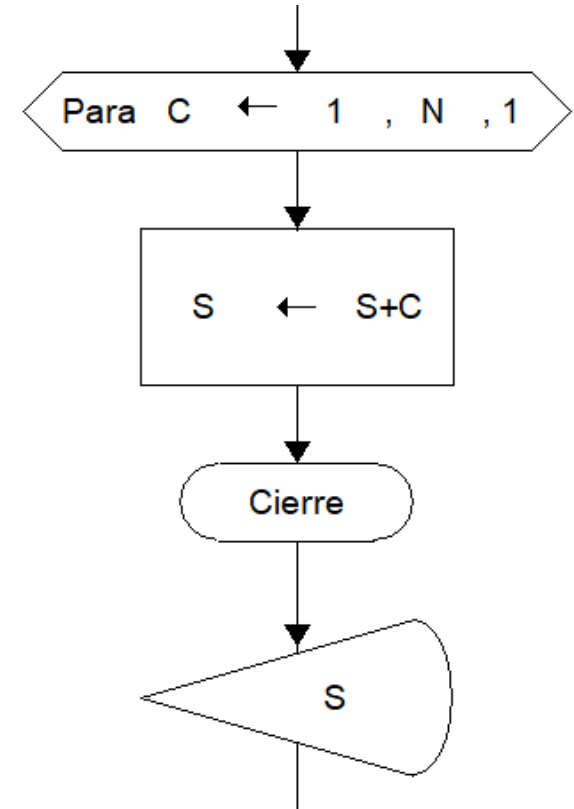
PROC 2

Fin_del For

PROC 3



Proceso que
se repite



En programación usamos el bucle **FOR** para repetir una o mas instrucciones un número determinado de veces, muy parecido al **WHILE**, solo que el bucle FOR es mas compacto en el código y de modo automático controla el número de iteraciones o pasos.

ESTRUCTURA DO WHILE (HACER MIENTRAS)

Hacer

PROC 1

PROC 2

Mientras condición

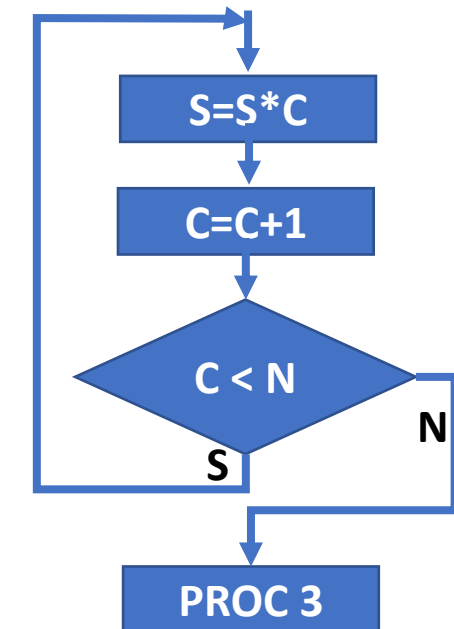
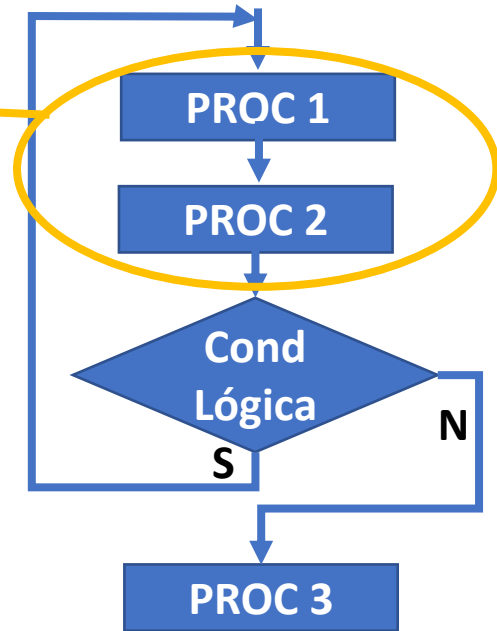
PROC 3

La estructura **DO WHILE** cumple la misma función que la estructura **WHILE**. La diferencia está en que la estructura **WHILE** **comprueba la condición al inicio** y **DO WHILE lo hace al final**. Es por ello que la estructura **DO WHILE** ejecuta por lo menos una vez los procesos.

Ej. Ingresar 10 números y muestre la sumatoria de ellos.

Cond. Lógica → **Repetir mientras contador < 10**

Proceso
que se
repite



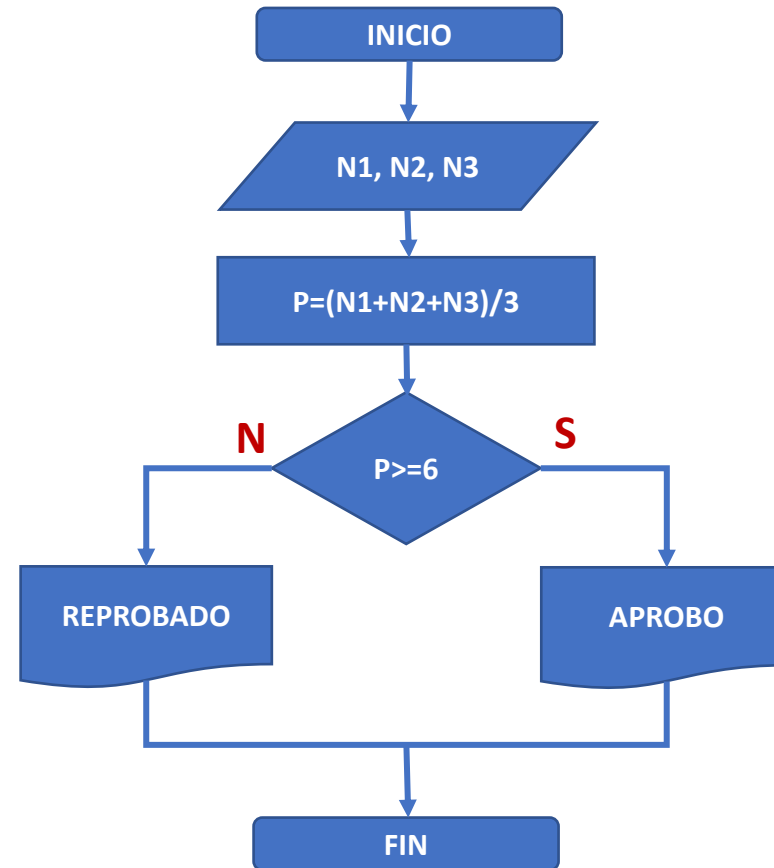
Dentro de los problemas que se pueden resolver con estas estructuras encontramos:

1. Problemas donde **se conoce la cantidad** de veces que se repite el ciclo.
Generalmente ***N***.
2. Problemas donde **NO se conoce** la cantidad de veces que se repite el ciclo.
3. Problemas de **generación de valores**.

¿CUÁNDO UTILIZAR UN BUCLE U OTRO?

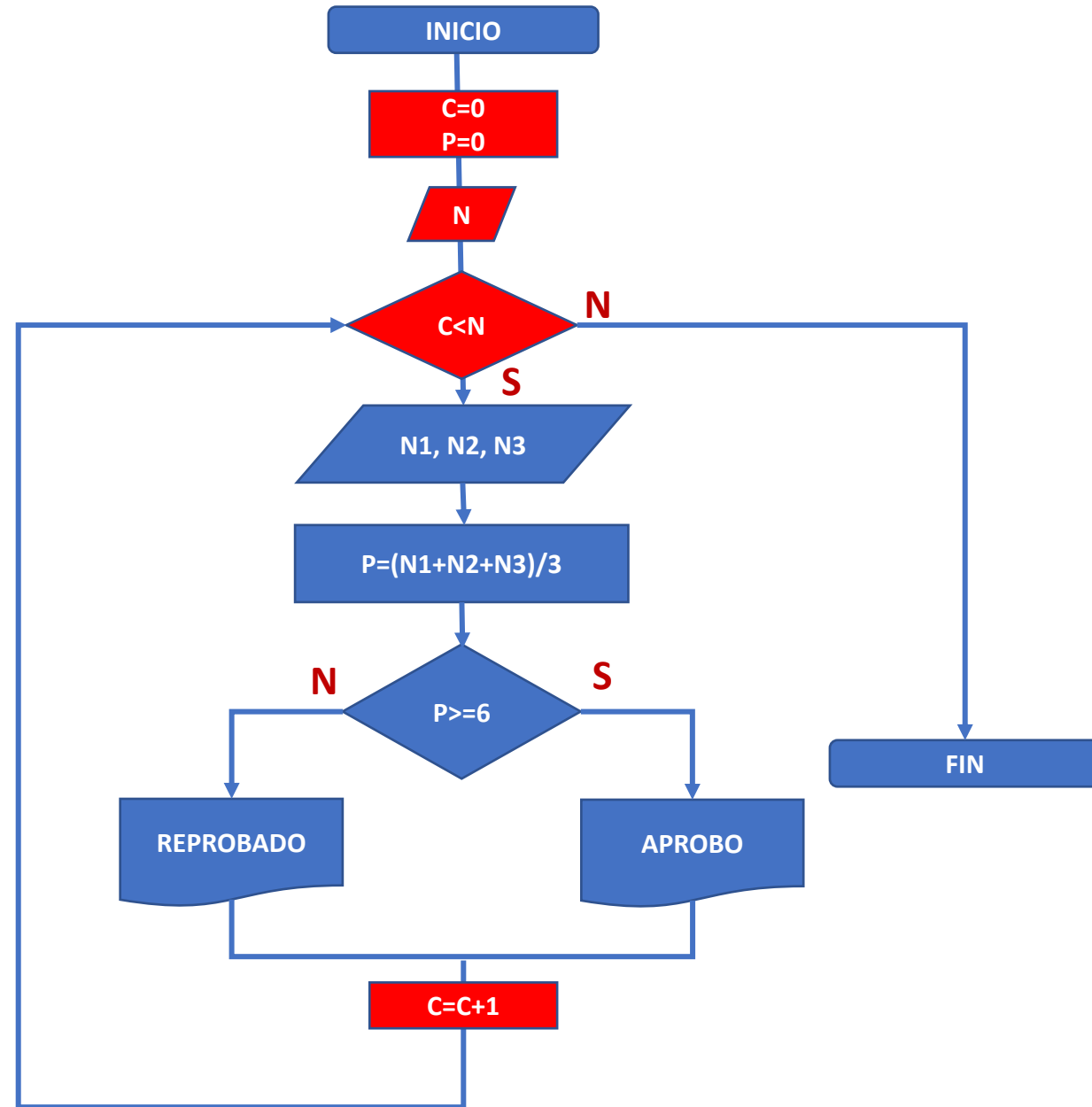
- La estructura **FOR** suele utilizarse cuando **se conoce** exactamente el número de iteraciones del bucle.
- La estructura **WHILE** suele utilizarse cuando **se conoce o No** el número de iteraciones del bucle, pudiendo ser éste mayor o igual a 0.
- La estructura **Do..While** suele utilizarse cuando **se conoce o No** el número de iteraciones del bucle, pudiendo ser éste mayor o igual a 1.

Por Ejemplo: Pedir las calificaciones de un alumno correspondientes a tres exámenes, obtener el respectivo promedio y determinar si dicho alumno aprobó o reprobó. Tomar 6 como nota mínima de aprobación.



Ahora, qué pasaría si serían N alumnos.

WHILE



Ahora, qué pasaría si se quisiera calcular el promedio general de todos los alumnos.

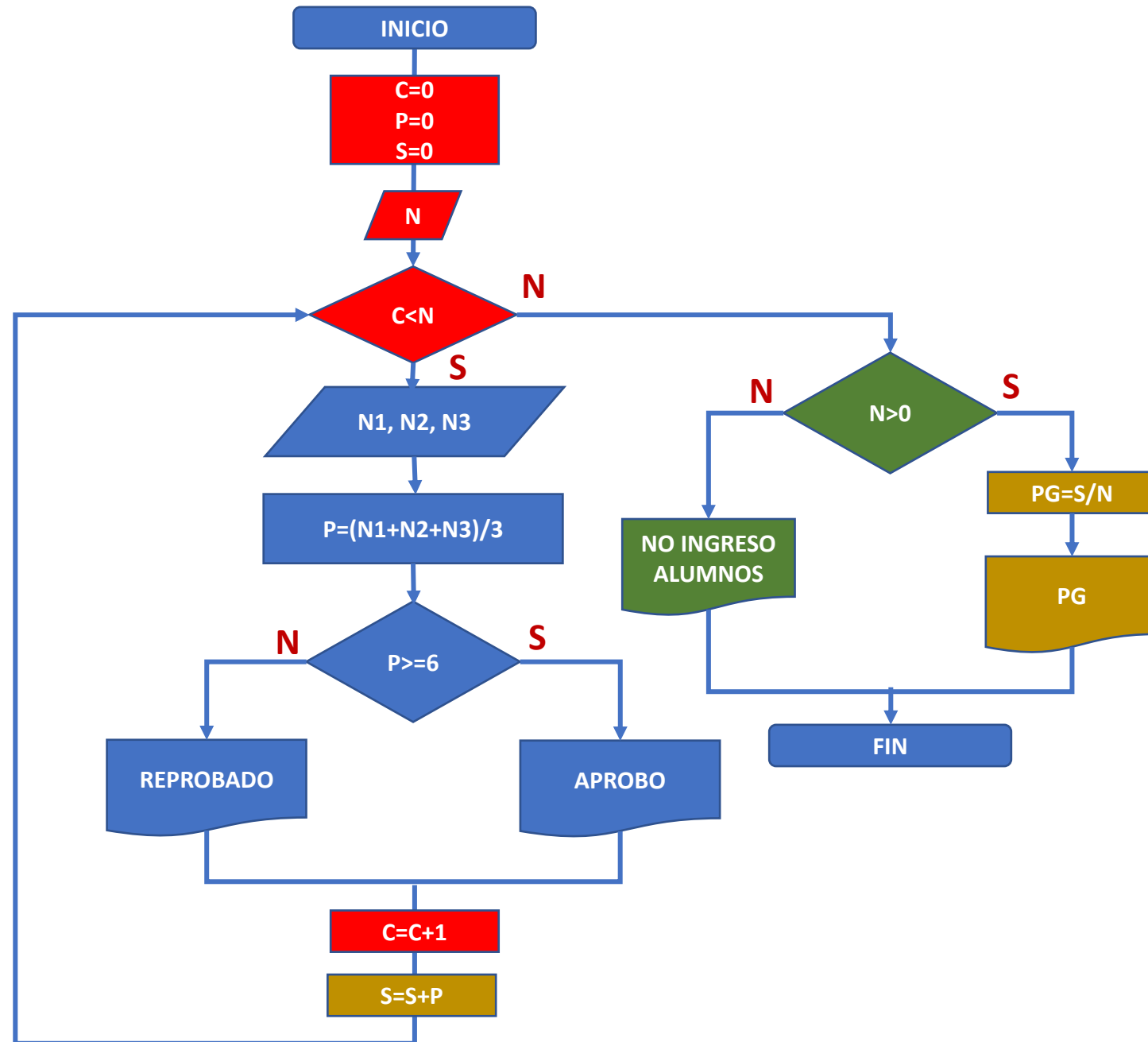


¿ ESTA RESUELTO EL PROBLEMA?

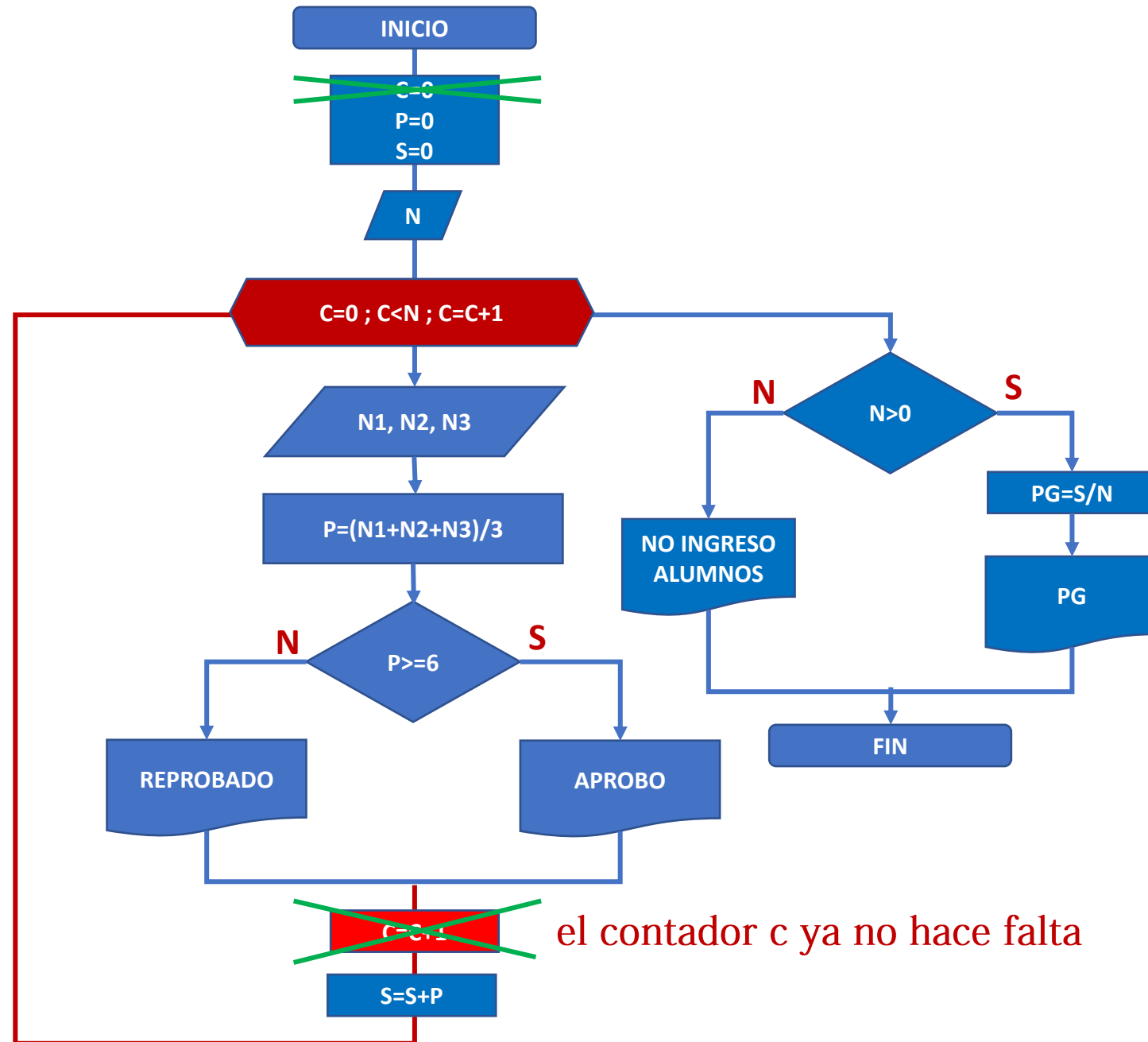
NO

¿ QUE PASA CUANDO N=0?

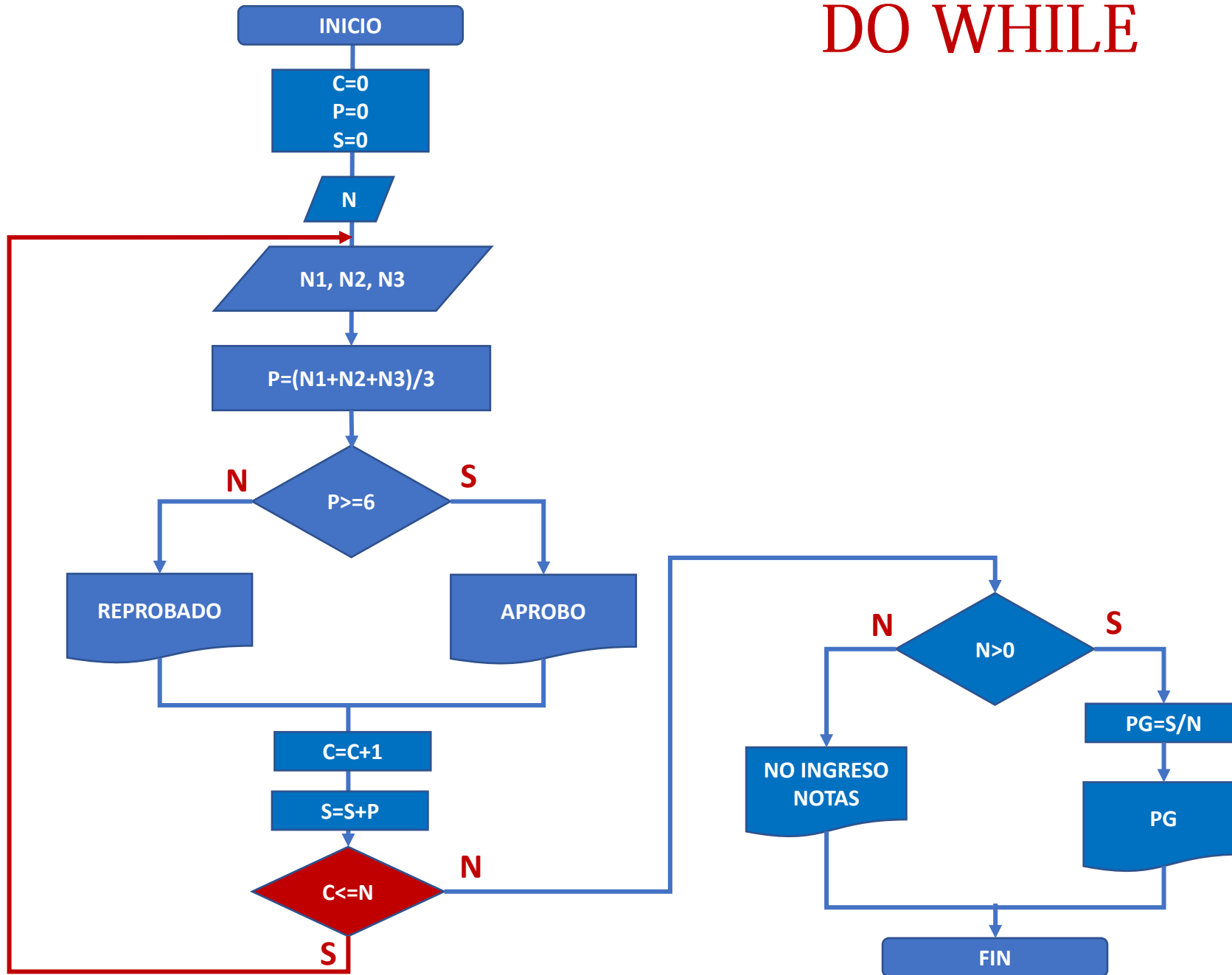
WHILE



FOR



DO WHILE



Ej.1 Generar la siguiente serie: 10,9,8,7,6,5,4,3,2,1,0

Ej. 2 Generar la tabla de multiplicar de un número X ingresado por el usuario del 1 al 12.

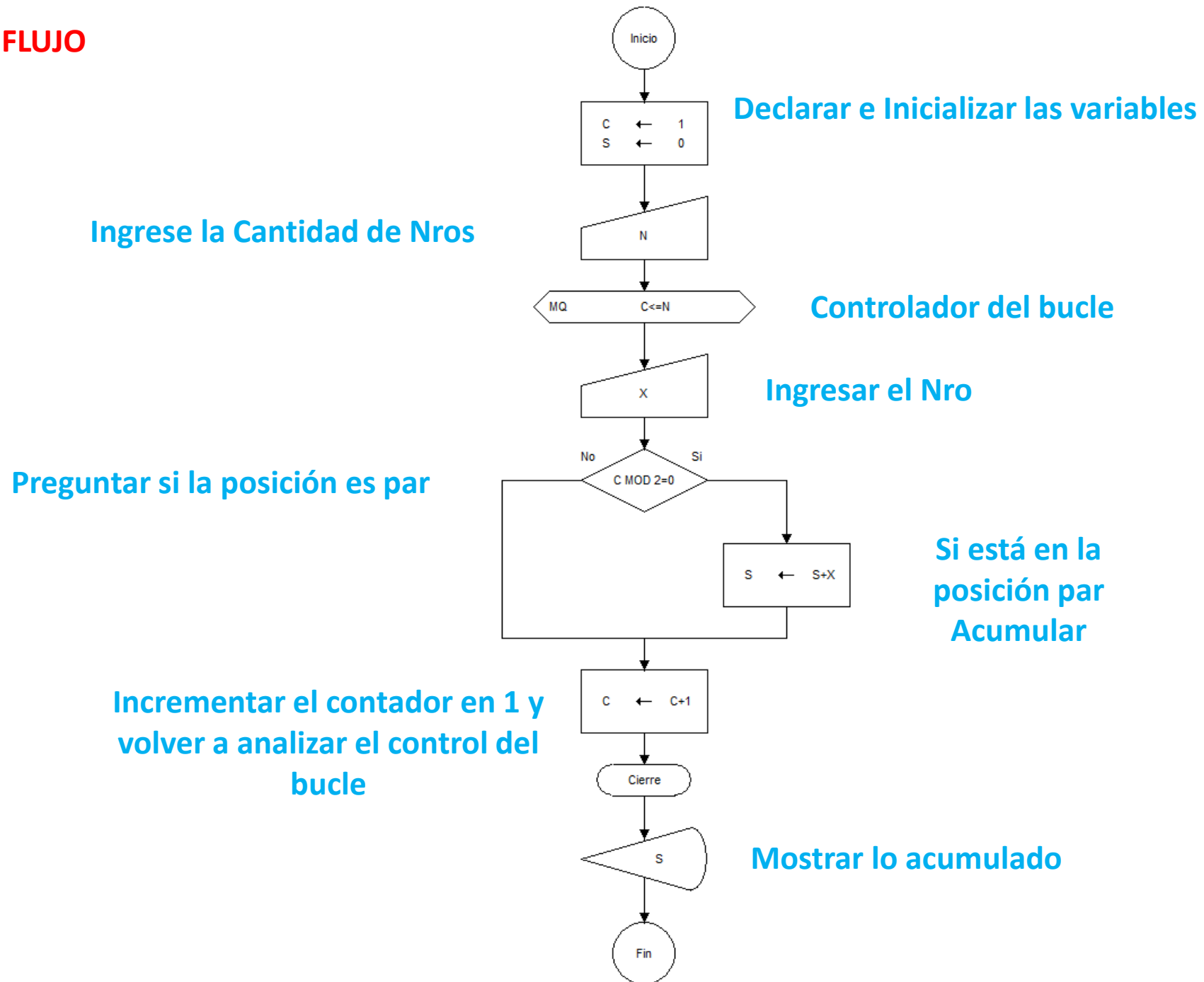
Ej. 3 Ingrese N valores y determine la suma de los valores ingresado en la posición par.

Ingrese N valores y determine la suma de los valores ingresado en la posición par.

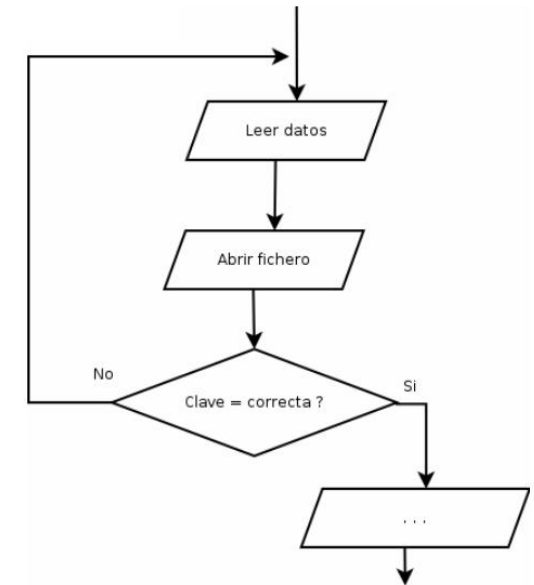
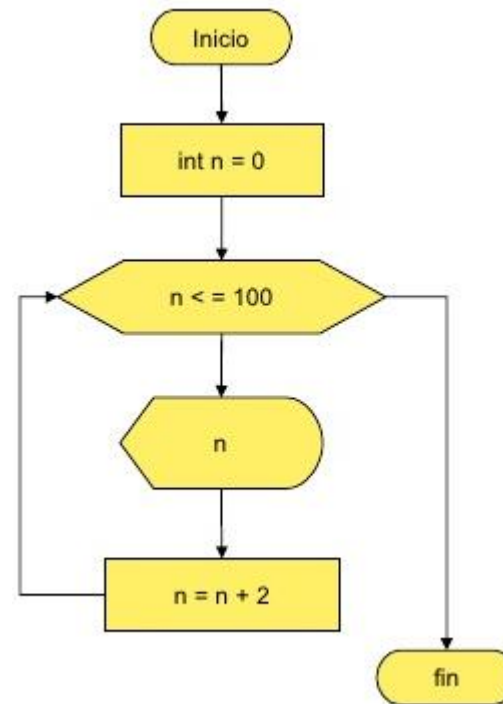
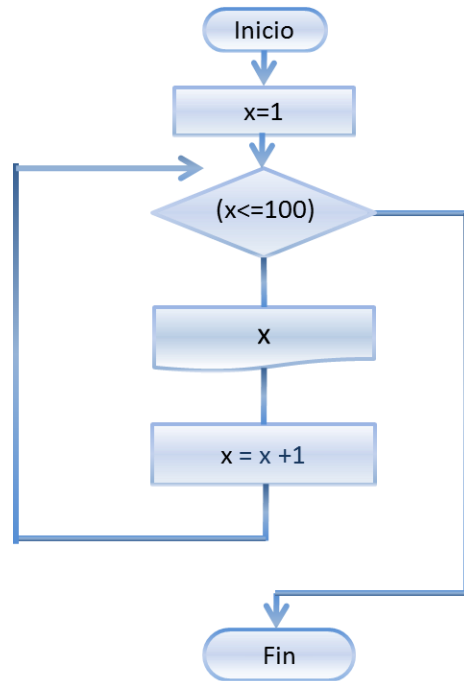
Ej. Si la cantidad de valores es **5** y los nros. Ingresados los siguientes **23, 4, 56, 45, 25**

Los valores que tendría que sumar serian: **4 y 45**, que están en la posición 2 y 4, dando como Resultado **49**

DIAGRAMA DE FLUJO



EJERCICIOS PARA AFIANZAR LOS TIPOS DE ESTRUCTURAS DE REPETICION



PROBLEMAS DONDE SE CONOCE LA CANTIDAD DE VECES QUE SE REPITE EL CICLO

Práctica 1:

Introduzca N números enteros y muestre el promedio de los mismos.

Práctica 2:

Introduzca N números enteros y muestre el porcentaje de números positivos ingresados.

Práctica 3:

Introduzca N valores y determine el mayor de ellos y en qué posición se encuentra.

Analice que pasaría si el valor ingresado para N es 0.

PROBLEMAS DONDE NO SE CONOCE LA CANTIDAD DE VECES QUE SE REPITE EL CICLO

Práctica 1:

Introduzca una cantidad no determinada de valores, cuyo final está determinado por el valor cero, y determine el promedio de ellos.

Práctica 2:

Dada una cantidad indeterminada de alumnos, ingrese las edades de los mismos, calcule y muestre el porcentaje de alumnos que estén entre 18 y 20 años.

Práctica 3:

Introduzca una cantidad indeterminada de números enteros, determine la sumatoria e indique si hubo algún número negativo.

PROBLEMAS DE GENERACIÓN DE VALORES

Práctica 1:

Genere los números impares menores a 100 y dé a conocer la suma de ellos.

Práctica 2:

Genere la tabla de multiplicar de un número X ingresado por el usuario del 1 al 10, desde un valor inicial hasta un valor final. [Por ejemplo la tabla del 5 desde 4 hasta el 15.](#)

Práctica 3:

Generar N números aleatorios de 2 dígitos y mostrar el porcentaje de los números pares. (Utilizar la función random(x)).

Práctica 4:

Realizar un menú que considere las siguientes opciones:

Caso 1: Cubo de un Número

Caso 2: Número Par o Impar

Caso 3: Salir

MAS PRACTICA SOBRE ESTRUCTURA REPETITIVAS

Ejercicio 1

Dar los valores de la raíces de la ecuación de segundo grado, ingresando los coeficientes a , b , c como datos. Considere todos los casos posibles, y mostrar también un mensaje indicando si son reales iguales, reales distintas, o complejas.

CON BUCLE conociendo N

Se ingresa N ternas de coeficientes y mostrar los valores de las raíces de la ecuación de segundo grado. Mostrar también con un mensaje indicando si son reales iguales, distintas o complejas.

CON BUCLE cuando no se conoce N

Ingresa una cantidad **indeterminada** de ternas correspondientes a los coeficientes de una ecuación de segundo grado. Se pide:

- a) Mostrar los valores de las raíces
- b) Indique con un mensaje si las raíces son iguales, distintas o complejas
- c) Muestre la cantidad de ternas analizadas
- d) Razone para realizar el fin del programa

Ejercicio 2

Se ingresan una cantidad de valores enteros no determinada. Se desea saber cuántos de los valores ingresados fueron pares y cuantos fueron impares. El ingreso termina cuando el valor ingresado es cero.

Ejercicio 3

El dueño de una remisería de N coches, desea que, ingresando la recaudación cada 30 días de un coche obtener cierta información por cada uno de los autos.

SE PIDE:

- Cuanto recauda en promedio por día.
- Determinar y mostrar cuál sería el sueldo del chofer, sabiendo que, si la recaudación mensual supera los 9000 pesos, el sueldo es el 30% de la recaudación realizada; y si no lo supera, el sueldo es el 25%.
- Si el promedio de lo recaudado por día es menor a \$300, muestre un mensaje indicando que dicho chofer se debe esforzar más.

FIN DE LA CLASE

Ejercicio N° 1:

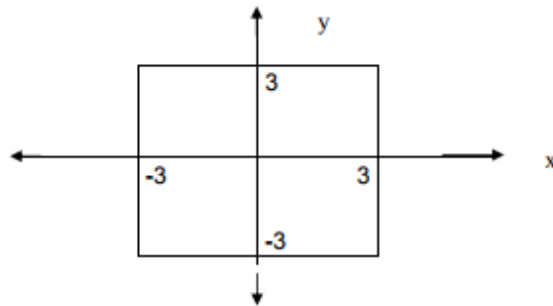
Ingresar el valor del Kw y el consumo eléctrico en una casa, se pide:

- Mostrar un mensaje "No tiene consumo" si el consumo es igual a cero.
- Mostrar lo que debe pagar el usuario, teniendo en cuenta que se le realiza un descuento del 5% para consumos menores a 350 Kw.

Ejercicio N° 2

Ingresar las coordenadas de un punto en el plano, dos variables X, Y.

Indicar si dicho punto si está dentro o fuera del cuadrado.



Ejercicio N° 1 - Resolver son Estructura Mientras

Para N empleados que trabajan en una fábrica ingrese por cada uno de ellos el valor de cada hora y la cantidad de horas trabajadas por un empleado, se pide mostrar por cada uno de ellos:

- Mostrar un mensaje "No trabajo" si la cantidad de horas es igual a cero.
- Mostrar el sueldo del empleado, teniendo en cuenta que se le realiza un descuento del 11% para jubilación y de un 3% para obra social, y si trabajó más de 60 hs tiene un premio de 5% sobre el total de cantidad de horas*valor de la hora

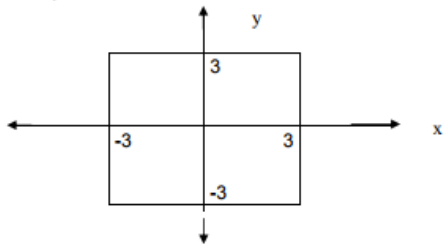
Y además el concepto total en pago de sueldos por la fábrica.

Ejercicio N° 2 - Resolver con Estructura Para

Ingrese N coordenadas de puntos en el plano, dos variables X, Y.

Indique :

- Si dicho punto se encuentra dentro o fuera del cuadrado.
- Cuántos puntos están dentro del cuadrado.
- Porcentaje de puntos fuera del cuadrado



Ejercicio N° 1

Por cada uno de los empleados que trabajan en una fábrica se ingresa por cada uno de ellos el valor de cada hora y la cantidad de horas trabajadas , se pide mostrar por cada uno de ellos:

- Mostrar un mensaje "No trabajo" si la cantidad de horas es igual a cero.
- Mostrar el sueldo del empleado, teniendo en cuenta que se le realiza un descuento del 11% para jubilación y de un 3% para obra social, y si trabajó más de 60 hs tiene un premio de 5% sobre el total de cantidad de horas*valor de la hora

Y además el concepto total en pago de sueldos por la fábrica.

NOTA: El final de datos se da ,cuando ingresa un valor de hora igual a 0 (cero).

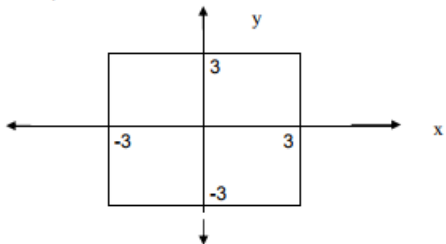
Ejercicio N° 2

Se ingresan coordenadas X, Y correspondiente a las coordenadas de puntos en el plano.

Para finalizar indique si se quieren seguir ingresando datos.

Indique :

- Si dicho punto se encuentra dentro o fuera del cuadrado.
- Cuántos puntos están dentro del cuadrado.
- Porcentaje de puntos fuera del cuadrado



Ejercicio N° 1

Para un empleado que trabaja en una fábrica se ingresa el valor de cada hora y la cantidad de horas trabajadas, se pide mostrar:

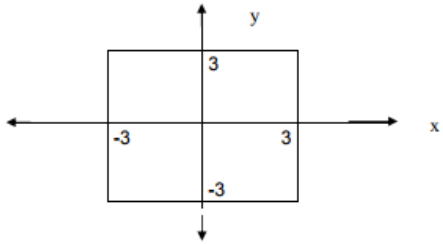
- Mostrar un mensaje “No trabajo” si la cantidad de horas es igual a cero.
- Mostrar el sueldo del empleado, teniendo en cuenta que se le realiza un descuento del 11% para jubilación y de un 3% para obra social, y si trabajó más de 60 hs tiene un premio de 5% sobre el total de cantidad de horas*valor de la hora

Ejercicio N° 2

Se ingresan las variables X, Y; correspondiente a sus coordenadas de un punto.

Indique :

- Si dicho punto se encuentra dentro o fuera del cuadrado.



Ejercicio N° 1 (Realizar con ciclo mientras)

Para N empleados que trabajan en una fábrica se ingresa el valor de cada hora y la cantidad de horas trabajadas, se pide mostrar:

- Mostrar un mensaje “No trabajo” si la cantidad de horas es igual a cero.
- Mostrar el sueldo del empleado, teniendo en cuenta que se le realiza un descuento del 11% para jubilación y de un 3% para obra social, y si trabajó más de 60 hs tiene un premio de 5% sobre el total de cantidad de horas*valor de la hora

Ejercicio N° 2 - (Realizar con ciclo Para)

Se ingresan N pares de valores para las variables X, Y; correspondiente a las coordenadas de un punto en el plano.

Indique :

- Si dicho punto se encuentra dentro o fuera del cuadrado.
- Cantidad de puntos fuera del cuadrado.

