



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Ing. García Morales Karina.

Asignatura: Fundamentos de programación.

Grupo: 22

No de Práctica(s): 4

Integrante(s): Alvarado Pérez Norma Laura.

Semestre: 2018-2

Fecha de entrega: 13/ Marzo / 2018

Observaciones:

CALIFICACIÓN: _____

Guía práctica de estudio 04: Diagramas de flujo.

Objetivo: Elaborar diagramas de flujo que representen soluciones algorítmicas vistas como una serie de acciones que comprendan un proceso.

Desarrollo:

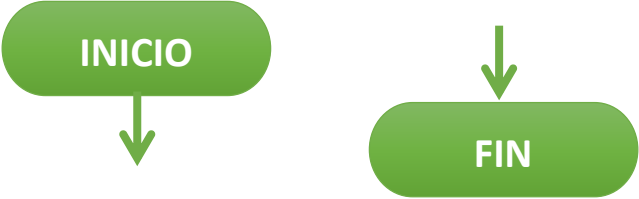
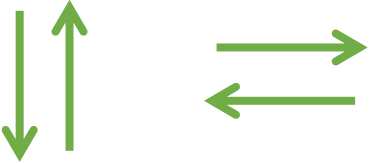


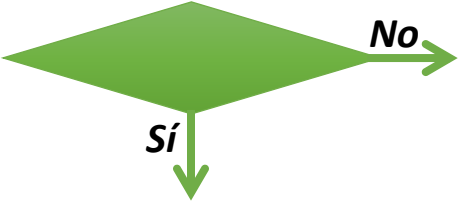

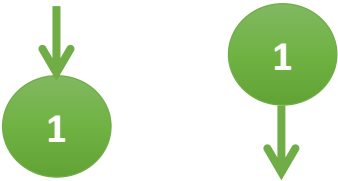
Un diagrama de flujo es la representación gráfica de un proceso, en las ciencias de la computación es la representación gráfica de un algoritmo.

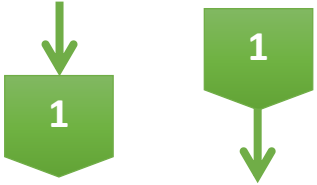


Reglas para diseñar un diagrama de flujo

- ➡ Todo diagrama tiene un inicio y un final (finito)
- ➡ Las líneas utilizadas para indicar dirección deben de ser rectas, solo verticales y horizontales.
- ➡ Todas las líneas utilizadas para indicar dirección deben llegar a un símbolo.
- ➡ A cada símbolo solo le puede llegar una línea de dirección.
- ➡ El diagrama se construye de arriba hacia abajo y de izquierda a derecha.
- ➡ La notación dentro del diagrama es independiente a un lenguaje de programación.

	Operador	Acción / Nombre
A R I T M É T I C O S	+	Sumar
	-	Restar
	*	Multiplicar
	/	Dividir
	^	Potencia
	mod	Módulo (residuo de la división)
R E L A C I O N A L E S	>	Mayor que
	<	Menor que
	>=	Mayor o igual que
	<=	Menor o igual que
	<>	Distinto
	=	Igual
	<-	Asignación
	()	Paréntesis

- ➡ Se debe usar notación camel para variables y nombres de funciones. Ejemplo: FuncionSuma ó variableSuma

Figura	Concepto
	<p>Inicio o Fin</p> <p>Se utilizan para marcar el Inicio y el Final de todo diagrama de flujo</p>
	<p>Dirección</p> <p>Se utilizan para indicar la dirección del flujo del diagrama.</p>
	<p>Datos de entrada.</p> <p>Se utiliza para indicar la lectura de datos.</p>
	<p>Proceso</p> <p>Se utiliza para indicar procesos, aquí se pueden hacer asignaciones u operaciones.</p>
	<p>Decisión</p> <p>Valida una condición, se debe indicar hacia dónde va cuando se cumple.</p>
	<p>Escritura</p> <p>Se utiliza para imprimir resultados.</p>
	<p>Conexión</p> <p>Se utiliza para indicar conexión dentro de la misma página.</p>

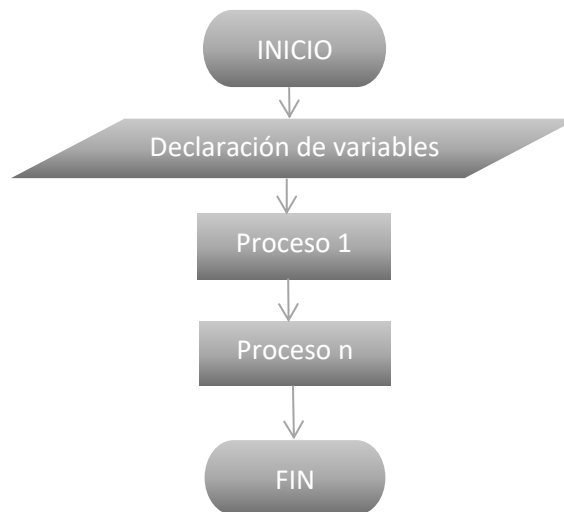
	<p>Conexión</p> <p>Se utiliza para indicar conexión con otra página.</p>
	<p>Módulo de un problema</p> <p>Llama a otros módulos o funciones.</p>
	<p>Decisión múltiple</p> <p>Almacena un selector que determina la rama por la que sigue el flujo.</p>

Estructuras de control de flujo



Secuencial.

Son las sentencias o declaraciones que se realizan en el orden en el que están escritas.

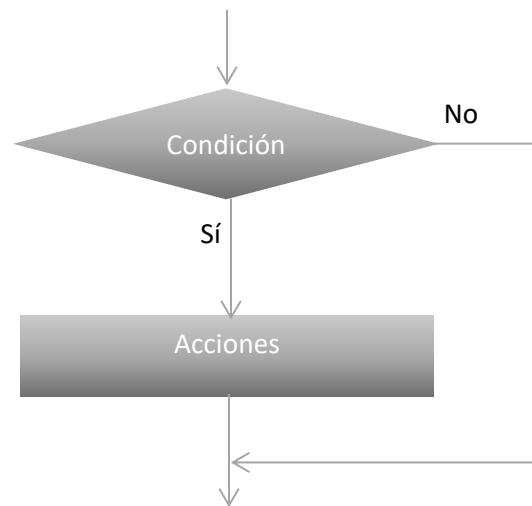




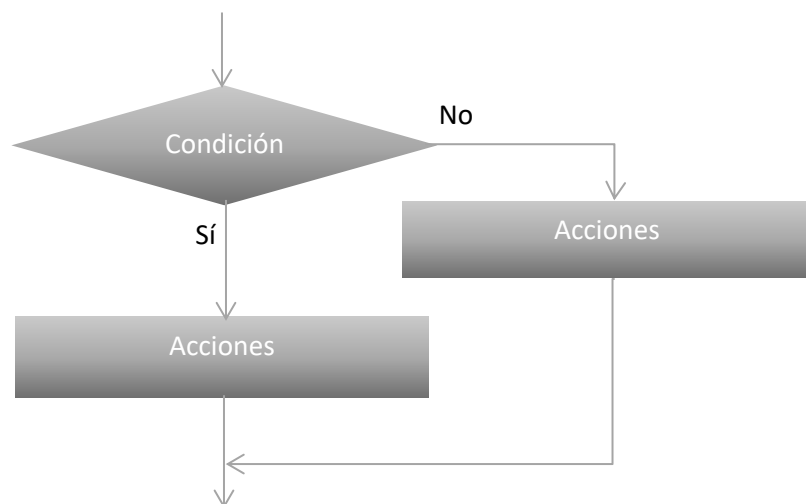
Condicional.

Permiten evaluar una expresión lógica (condición que puede ser verdadera o falsa) y dependiendo el resultado es el flujo de instrucciones siguiente.

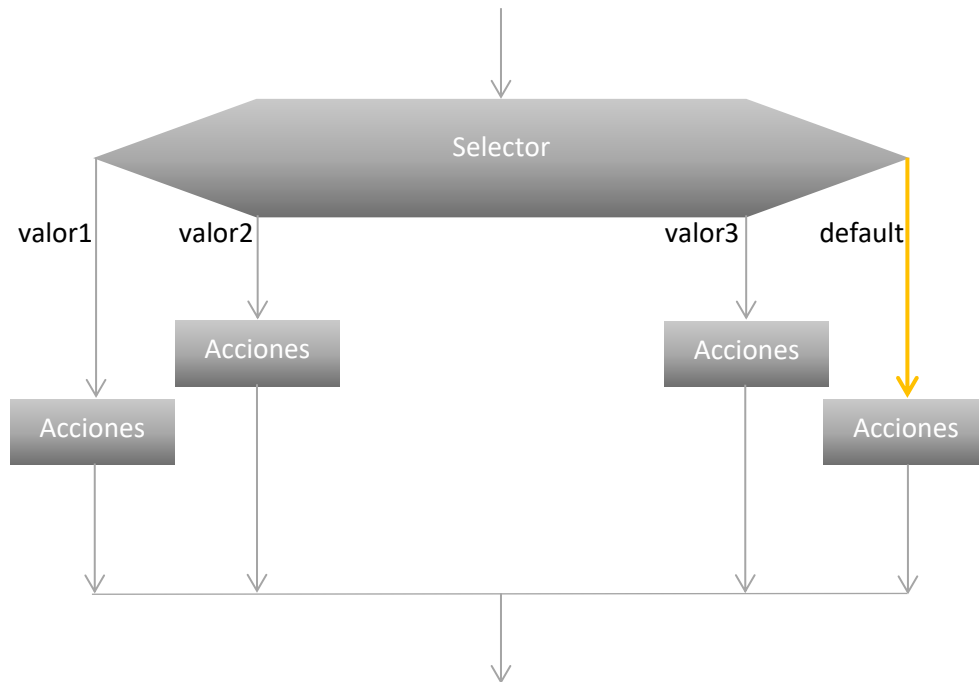
Condicional simple: Evalúa una condición y si ésta se cumple ejecuta un bloque de procesos



Condicional compuesta: Evalúa una condición y si ésta se cumple ejecuta un bloque de procesos, pero además tiene otro bloque en caso de que no se cumpla la condición.



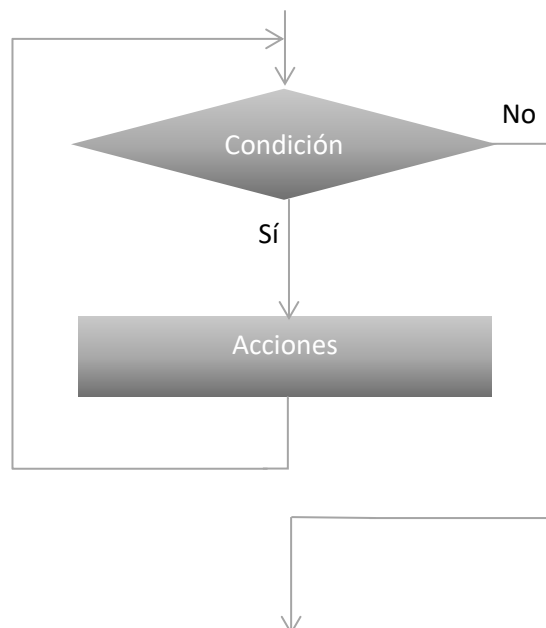
Condicional múltiple: Compara un selector con varias opciones, si se cumple con alguna realiza las acciones correspondientes, siempre cuenta con “default” que es el caso que se realizará cuando el selector no sea idéntico a cualquiera de las opciones o casos.



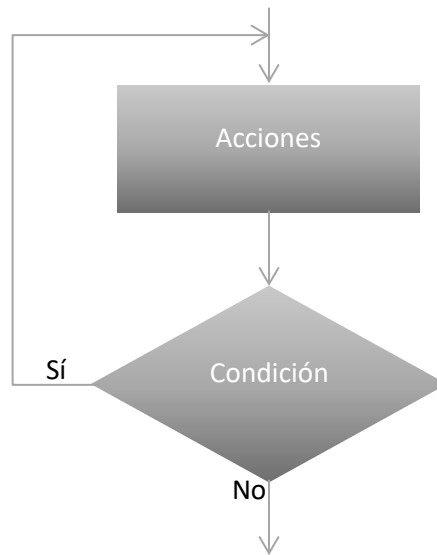
Iterativa.

Las estructuras iterativas o repetitivas, también llamadas cíclicas, permiten realizar un bloque de acciones mientras una expresión lógica sea verdadera.

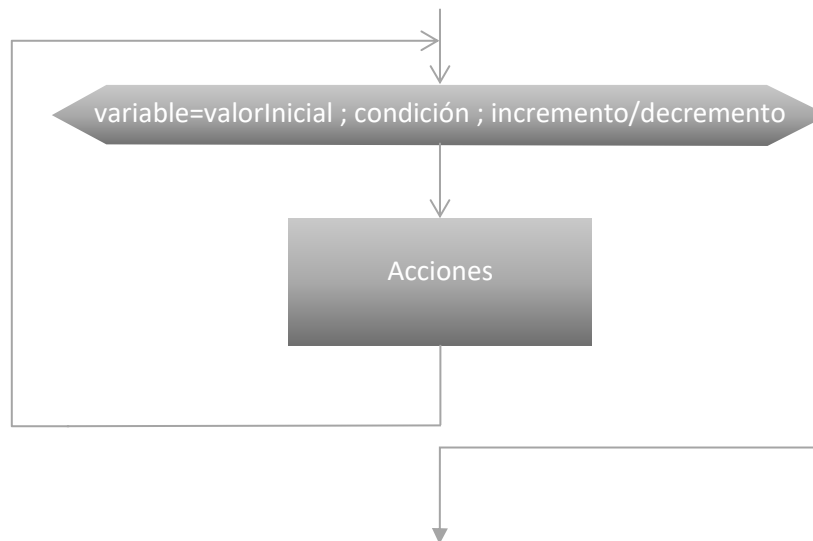
Estructura MIENTRAS: Se caracteriza por primero evaluar la expresión lógica (condición), si ésta es verdadera procede a realizar las acciones correspondientes hasta que la condición deje de ser verdadera.



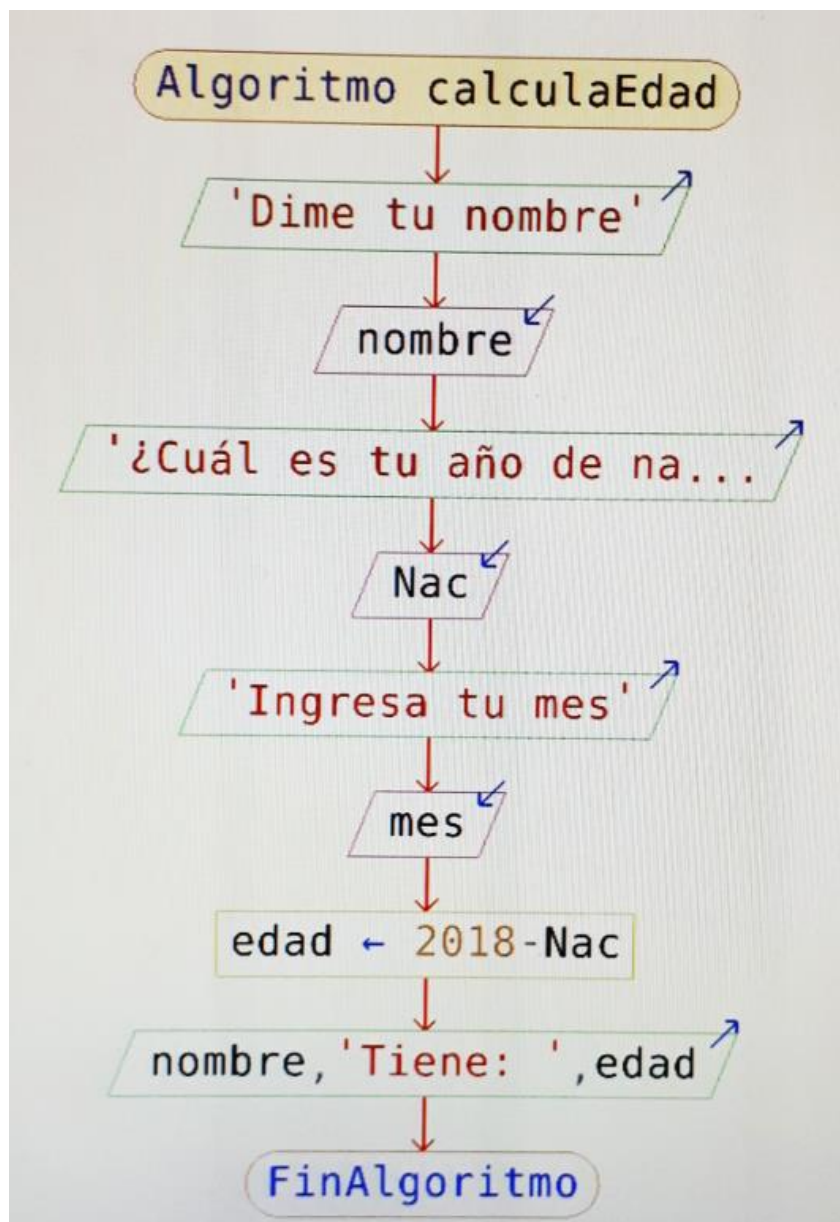
Estructura HACER-MIENTRAS: Se caracteriza por que realiza al menos una vez el proceso, después evalúa la expresión lógica y si es verdadera vuelve a ejecutar el código.



Estructura PARA: Dentro de su estructura tiene una variable contadora que es la que se compara, cuenta con un límite inferior y un límite superior, se realizan las acciones hasta el momento en que la variable contadora sobre pase el límite superior y deje de cumplir con la condición.



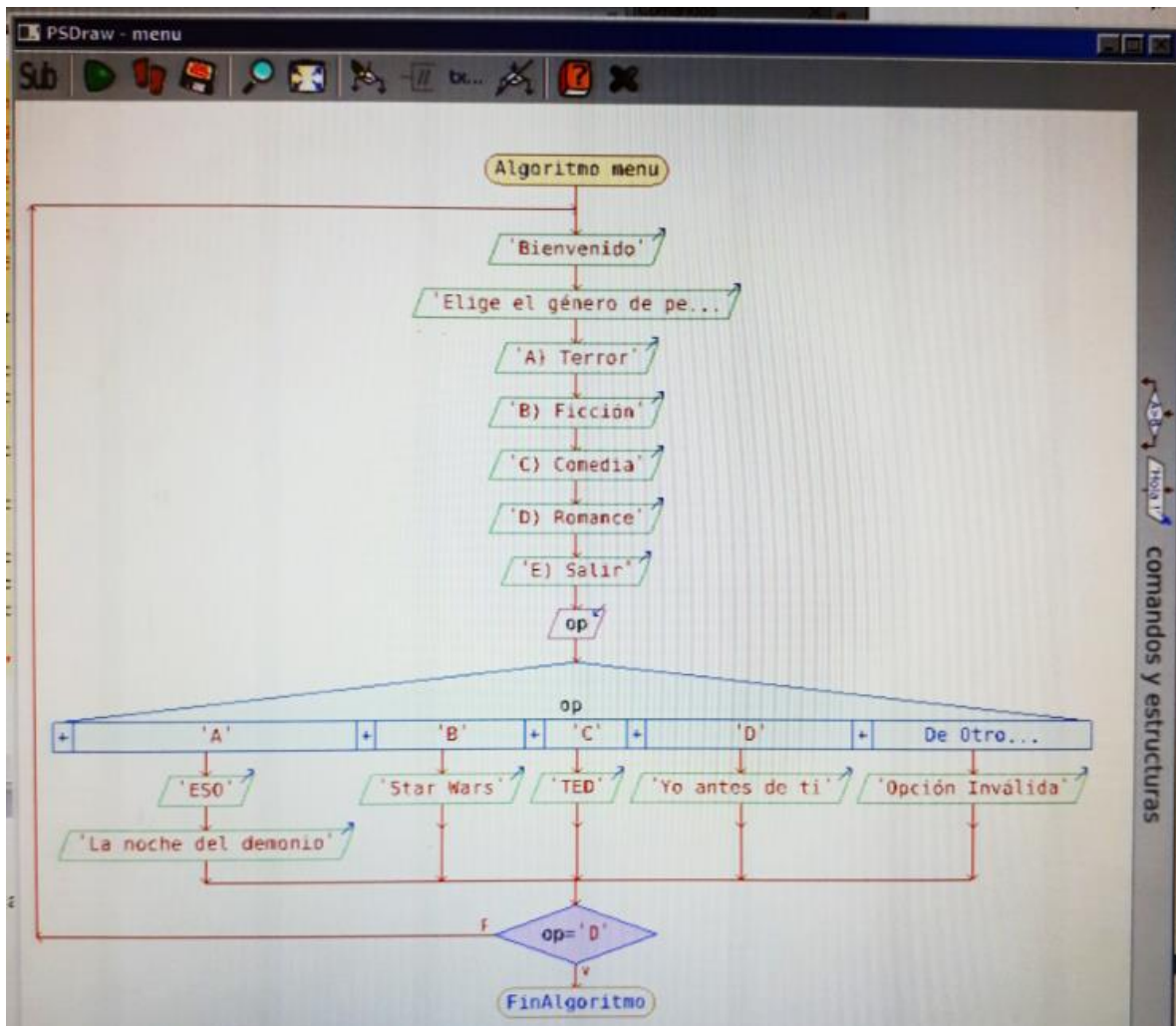
Ejercicios en el laboratorio



```
1 Algoritmo calculaEdad
2   Escribir 'Dime tu nombre'
3   Leer nombre
4   Escribir '¿Cuál es tu año de nacimiento?'
5   Leer Nac
6   Escribir 'Ingresa tu mes'
7   Leer mes
8   edad ← 2018 - Nac
9   Escribir nombre, 'Tiene: ', edad
10  FinAlgoritmo
11
12
```

PSeInt - Ejecutando proceso CALCULAEDAD

```
*** Ejecución Iniciada. ***
Dime tu nombre
> Laura
¿Cuál es tu año de nacimiento?
> 1991
Ingresa tu mes
> septiembre
LauraTiene: 27
*** Ejecución Finalizada. ***
```

```

1  Algoritmo menu
2  Repetir
3      Escribir 'Bienvenido'
4      Escribir 'Elige el género de película'
5      Escribir 'A) Terror'
6      Escribir 'B) Ficción'
7      Escribir 'C) Comedia'
8      Escribir 'D) Romance'
9      Escribir 'E) Salir'
10     Leer op
11     Segun op Hacer
12         'A':
13             Escribir 'ESO'
14             Escribir 'La noche del demonio'
15         'B':
16             Escribir 'Star Wars'
17         'C':
18             Escribir 'TED'
19         'D':
20             Escribir 'Yo antes de ti'
21         De Otro Modo:
22             Escribir 'Opción Inválida'
23     FinSegun
24     Hasta Que op = 'D'
25 FinAlgoritmo
  
```

PSInt - Ejecutando proceso MENU

```

*** Ejecución Iniciada. ***
Bienvenido
Elige el género de película
A) Terror
B) Ficción
C) Comedia
D) Romance
E) Salir
> a
Opción Inválida
  
```

caso default

```

Bienvenido
Elige el género de película
A) Terror
B) Ficción
C) Comedia
D) Romance
E) Salir
> A
ESO
La noche del demonio
  
```

caso 'A'

```

Bienvenido
Elige el género de película
A) Terror
B) Ficción
C) Comedia
D) Romance
E) Salir
> |
  
```

línea 10 instrucción 1

PSeInt

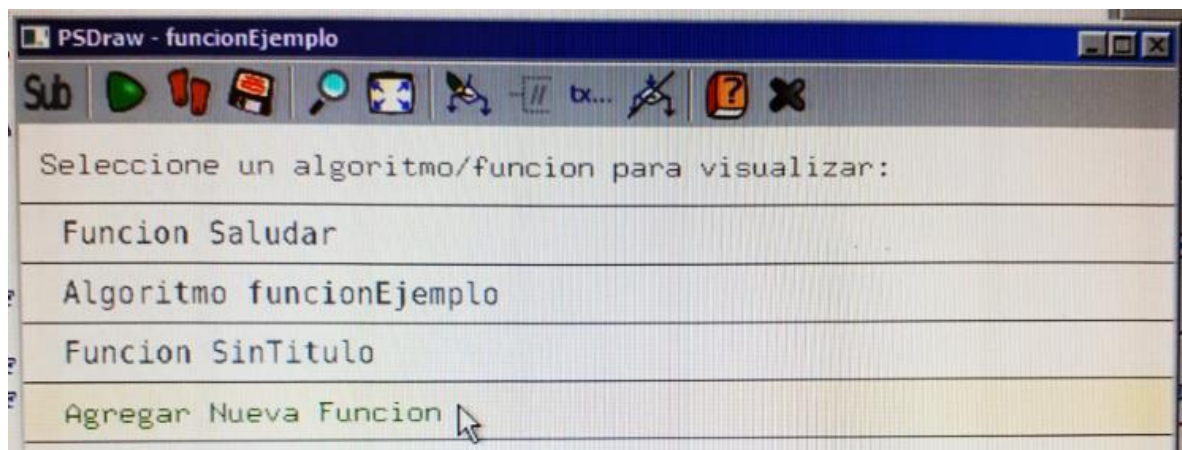
Archivo Editor Configurar Ejecutar Ayuda

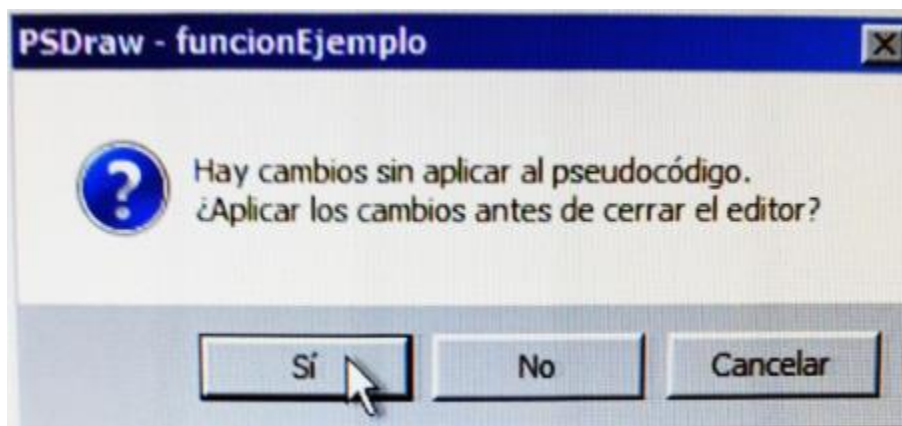
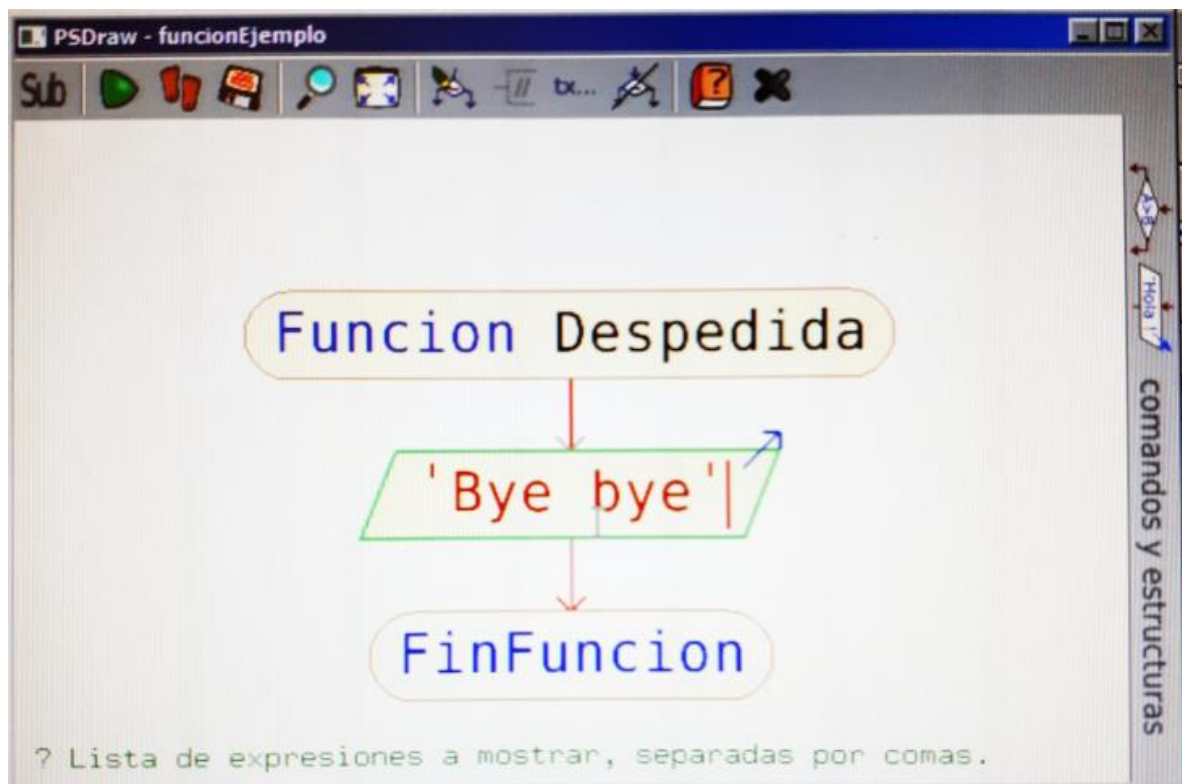
<sin_titulo>* <sin_titulo>* <sin_titulo>* X

```
1 // Las funciones van antes que el algoritmo principal
2 Funcion Saludar
3     Escribir 'Hola Mundo!'
4     Escribir 'Ejercicio de lau'
5     Escribir 'Regresa texto'
6 FinFuncion
7
8 Algoritmo funcionEjemplo
9     // texto prueba47589642
10    Escribir 'Llamar a función saludar'
11    Saludar // Aquí se mandó a llamar a la función
12
13 FinAlgoritmo
14
15 Funcion SinTitulo
16 FinFuncion
```

Lista de Variables

Operadores y Funciones





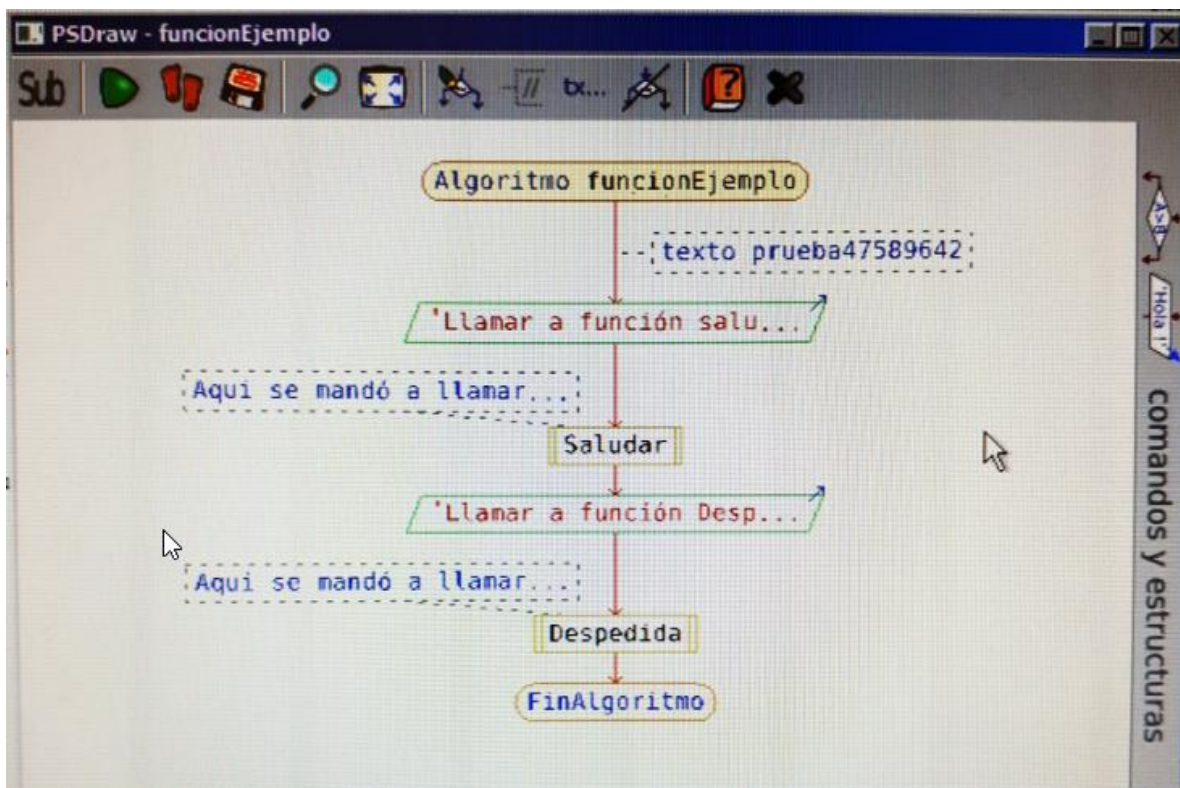
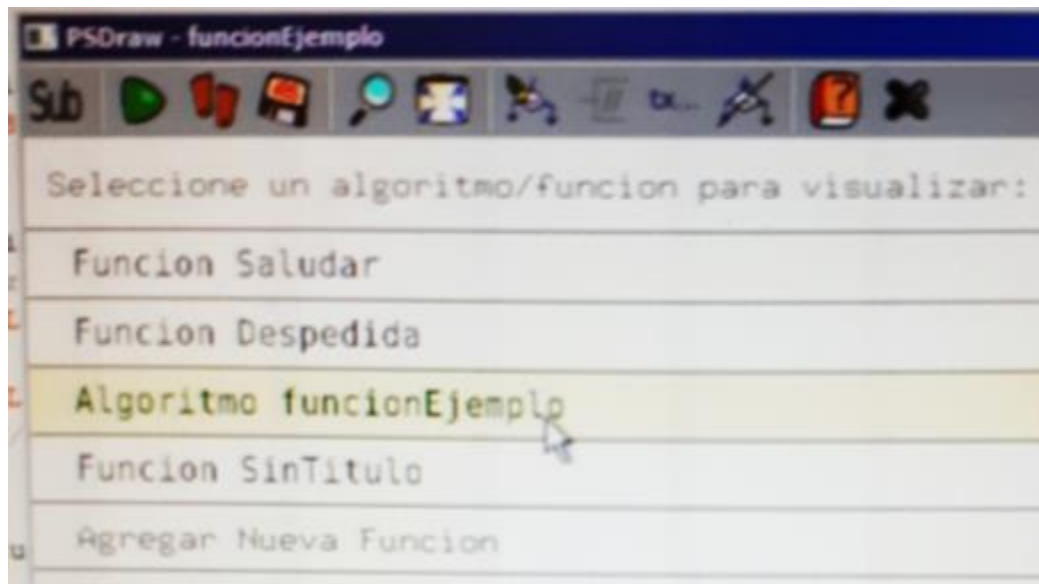
PSeInt

Archivo Editar Configurar Ejecutar Ayuda

<sin_titulo>* <sin_titulo>* <sin_titulo>* X

```
1 // Las funciones van antes que el algoritmo principal
2 Funcion Saludar
3     Escribir 'Hola Mundo!'
4     Escribir 'Ejercicio de lau'
5     Escribir 'Regresa texto'
6 FinFuncion
7
8 Algoritmo funcionEjemplo
9     // texto prueba47589642
10    Escribir 'Llamar a función saludar'
11    Saludar // Aqui se mandó a llamar a la función
12 FinAlgoritmo
13
14 Funcion SinTitulo
15 FinFuncion
16
17 Funcion Despedida
18     Escribir 'Bye bye'
19 FinFuncion
```

```
1 // Las funciones van antes que el algoritmo principal
2 Funcion Saludar
3     Escribir 'Hola Mundo!'
4     Escribir 'Ejercicio de lau'
5     Escribir 'Regresa texto'
6 FinFuncion
7
8 Funcion Despedida
9     Escribir 'Bye bye'
10 FinFuncion
11
12 Algoritmo funcionEjemplo
13     // texto prueba47589642
14     Escribir 'Llamar a función saludar'
15     Saludar // Aqui se mandó a llamar a la función
16     Escribir 'Llamar a función Despedida'
17     Despedida // Aqui se mandó a llamar a la función
18 FinAlgoritmo
19
20 Funcion SinTitulo
21 FinFuncion
```



```
PSInt - Ejecutando proceso FUNCIONEJEMPLO

*** Ejecución Iniciada. ***
Llamar a función saludar
Hola Mundo!
Ejercicio de lau
Regresa texto
Llamar a función Despedida
Bye bye
*** Ejecución Finalizada. ***
```

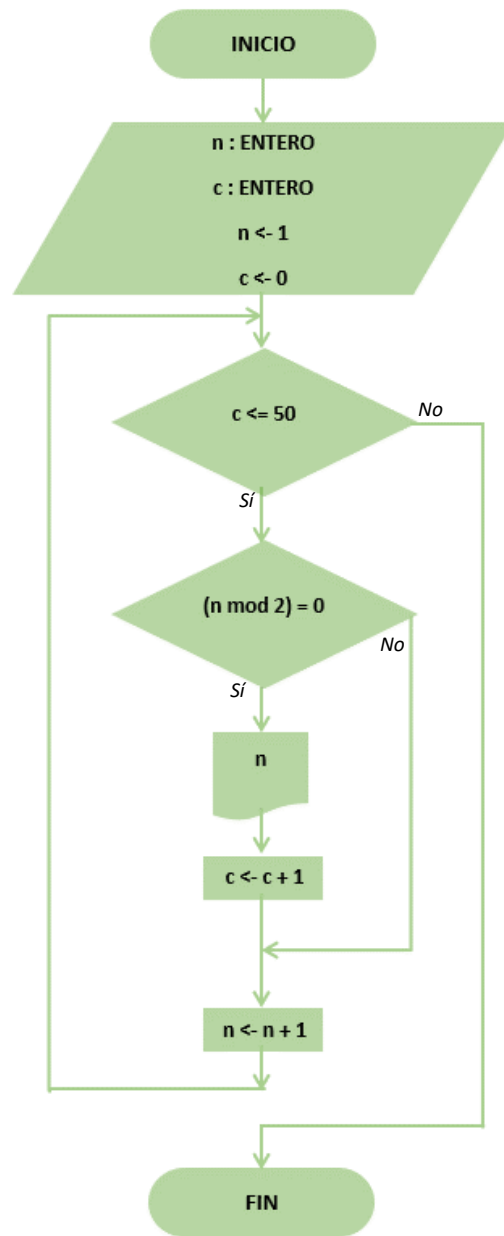
Ejercicios Propuestos

💚 PROBLEMA: Realiza un algoritmo que muestre los primeros 50 números pares

Análisis del problema:

- *DATOS DE ENTRADA*: El conjunto de números Naturales.
- *DATOS DE SALIDA*: Los primeros 50 números pares.
- *RESTRICCIÓN*: el cero no cuenta como par

Algoritmo (gráfico)

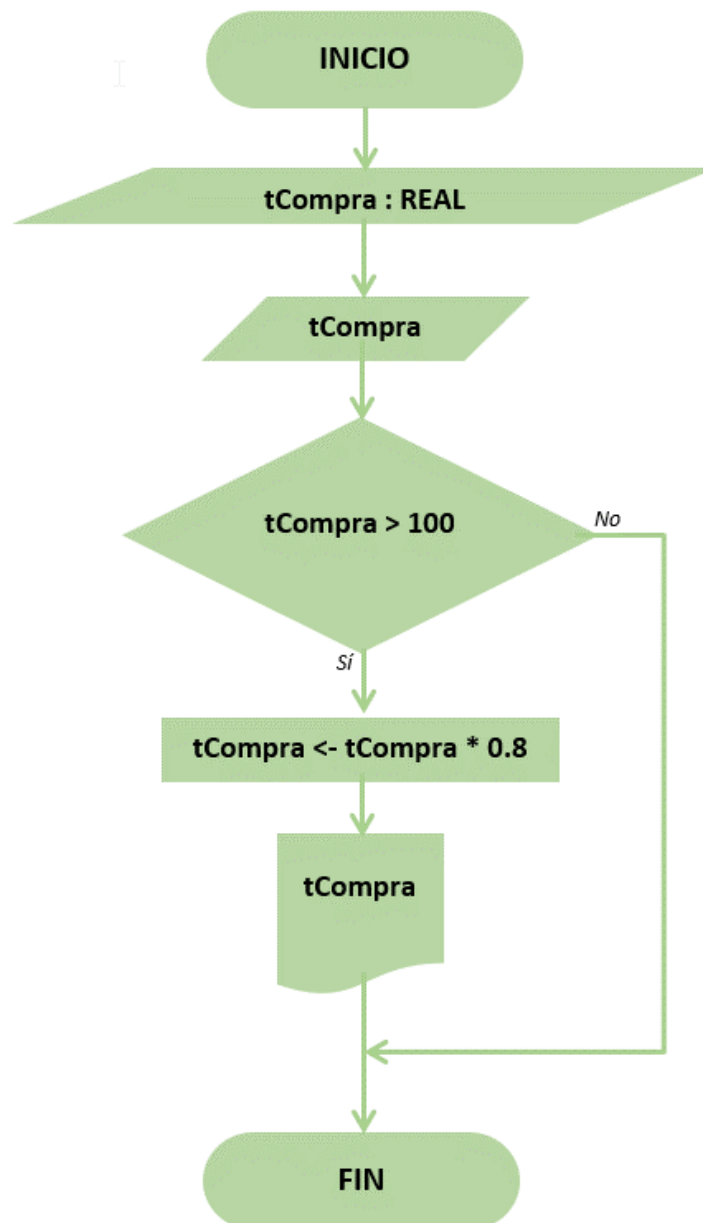


🍀 PROBLEMA: Realiza un algoritmo para una tienda, que ofrece 20% de descuento en compras mayores a \$100, el cliente introduce la cantidad comprada.

Análisis del problema:

- *DATOS DE ENTRADA*: El total de la compra realizada por el cliente
- *DATOS DE SALIDA*: Un número que es el 80% del total de la compra.

Algoritmo (gráfico)

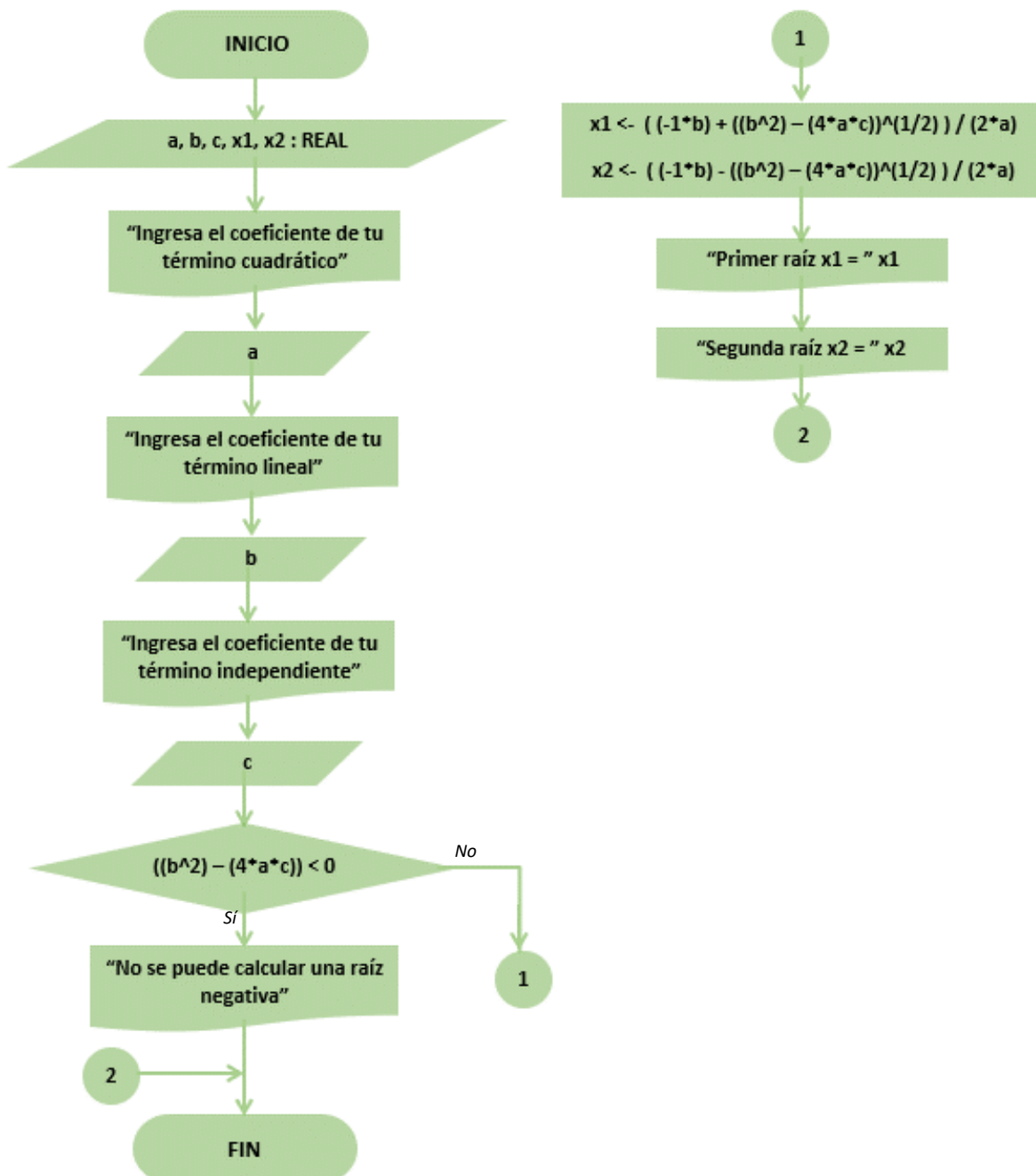


🌱 PROBLEMA: Realiza un algoritmo para obtener las posibles raíces de una ecuación cuadrática (fórmula de la "chicharronera").

Análisis del problema:

- *DATOS DE ENTRADA*: Los coeficientes de una ecuación cuadrática.
- *DATOS DE SALIDA*: Las posibles raíces de la ecuación cuadrática.
- *RESTRICCIONES*: El discriminante no puede ser menor a cero

Algoritmo (gráfico)



Conclusiones

- ➡ Logré identificar las estructuras en los diagramas de flujo y las utilicé para representar algoritmos siguiendo las reglas de su escritura para una mejor comprensión.