



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de Docencia

Laboratorios de Computación Salas A y B

Profesor: **Ing. Karina García Morales**

Asignatura: **Fundamentos de Programación (L)**

Grupo: **22**

No. de práctica(s): **Práctica 4. Diagramas de flujo.**

Integrante(s): **Jonathan Enrique Álvarez Hernández**

No. de lista o brigada: **2**

Semestre: **2026-1**

Fecha de entrega: **Martes 16 de Septiembre de 2025**

Observaciones:

CALIFICACIÓN: _____

Diagramas de flujo

Objetivo:

- El alumno elaborará diagramas de flujo que representen soluciones algorítmicas vistas como una serie de acciones que comprendan un proceso.

Desarrollo de la práctica:

Un diagrama de flujo es la representación gráfica de un proceso, es decir, muestra gráficamente el flujo de acciones a seguir para cumplir con una tarea específica.

Dentro de las ciencias de la computación, un diagrama de flujo es la representación gráfica de un algoritmo.



Reglas de construcción:








Son 9 las reglas que se tiene que seguir para la construcción de un diagrama de flujo:

1. Todo diagrama debe tener un inicio y un fin.
2. Las líneas utilizadas para indicar la dirección del flujo del diagrama deben ser rectas y verticales u horizontales, exclusivamente.
3. Todas las líneas utilizadas para indicar la dirección del flujo del diagrama deben estar conectadas a un símbolo.
4. El diagrama debe ser construido de arriba hacia abajo (top-down) y de izquierda a derecha (left to right).
5. La notación utilizada en el diagrama de flujo debe ser independiente del lenguaje de programación en el que se va a codificar la solución.
6. Se recomienda poner comentarios que expresen o ayuden a entender un bloque de símbolos.
7. Si la extensión de un diagrama de flujo ocupa más de una página, es necesario utilizar y numerar los símbolos adecuados.
8. A cada símbolo solo le puede llegar una línea de dirección de flujo.
9. Para nombrar variables y nombres de funciones se debe hacer uso de la notación de camello.

Símbolos:

Los diagramas de flujo poseen símbolos que permiten estructurar la solución de un problema de manera gráfica. Por tanto, es fundamental conocer los elementos que conforman este lenguaje gráfico.

	Representa el inicio o el fin del diagrama de flujo.
	Datos de entrada. Expresa lectura de datos.

	Proceso. En su interior se expresan asignaciones u operaciones.
	Decisión. Valida una condición y toma uno u otro camino.
	Escritura. Escribe el o los resultado(s).
	Conexión dentro de la misma página.
	Conexión entre diferentes páginas.
	Módulo de un problema. Llamada a otros módulos o funciones.
	Decisión múltiple. Almacena un selector que determina la rama por la que sigue el flujo.

Estructuras de control de flujo:

Las estructuras de control de flujo permiten tanto la ejecución condicional como la repetición de un conjunto de instrucciones.

Existen 3 estructuras de control: secuencial, condicional y repetitivas o iterativas.

- Estructura de control secuencial: Las estructuras de control secuenciales son las sentencias o declaraciones que se realizan una a continuación de otra en el orden en el que están escritas.

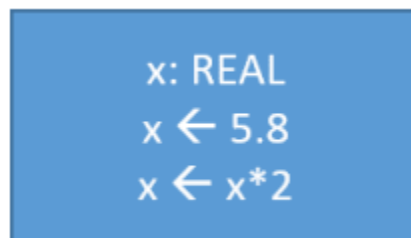


Figura 1. Ejemplo de estructura de control secuencial (Facultad de Ingeniería, 2025).

- Estructuras de control condicionales: Las estructuras de control condicionales permiten evaluar una expresión lógica (condición que puede ser verdadera o falsa) y, dependiendo del resultado, se realiza uno u otro flujo de instrucciones. Estas estructuras son mutuamente excluyentes (o se realiza una acción o se realiza la otra).

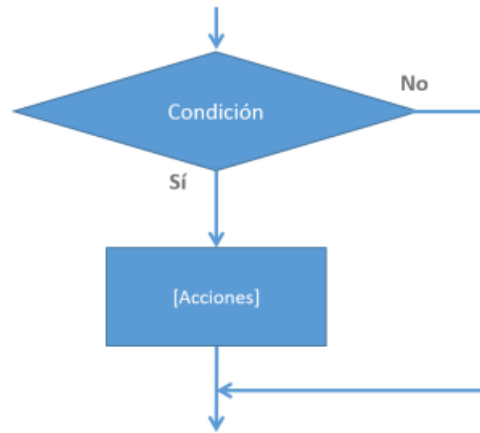


Figura 2. Ejemplo de estructura de control condicional simple (Facultad de Ingeniería, 2025).

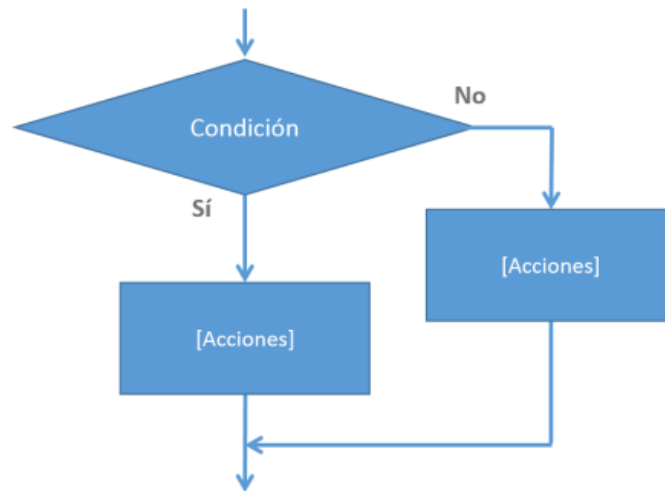


Figura 3. Ejemplo de estructura de control condicional doble (Facultad de Ingeniería, 2025).

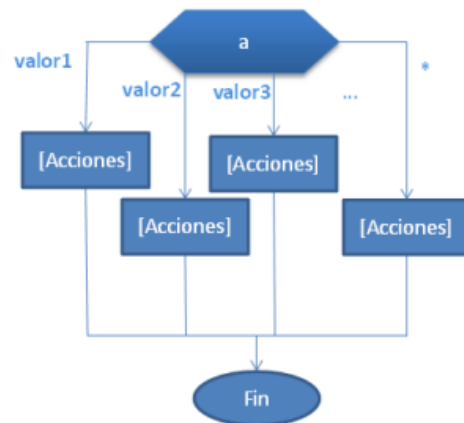


Figura 4. Ejemplo de estructura de control condicional múltiple (Facultad de Ingeniería, 2025).

- Estructuras de control iterativas: Las estructuras de control de flujo iterativas o repetitivas (también llamadas cíclicas) permiten realizar una serie de instrucciones mientras se cumpla la expresión lógica. Existen tres tipos de expresiones cíclicas MIENTRAS, HACER- MIENTRAS y HACER PARA.

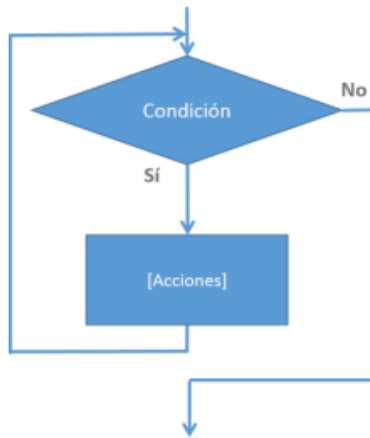


Figura 5. Ejemplo de estructura de control iterativa MIENTRAS (Facultad de Ingeniería, 2025).

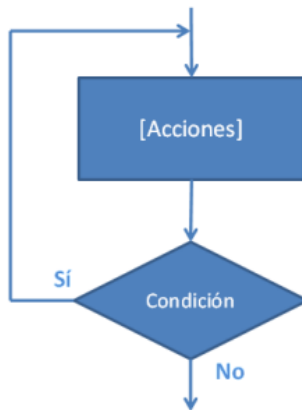


Figura 6. Ejemplo de estructura de control iterativa HACER-MIENTRAS (Facultad de Ingeniería, 2025).

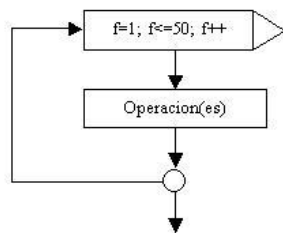
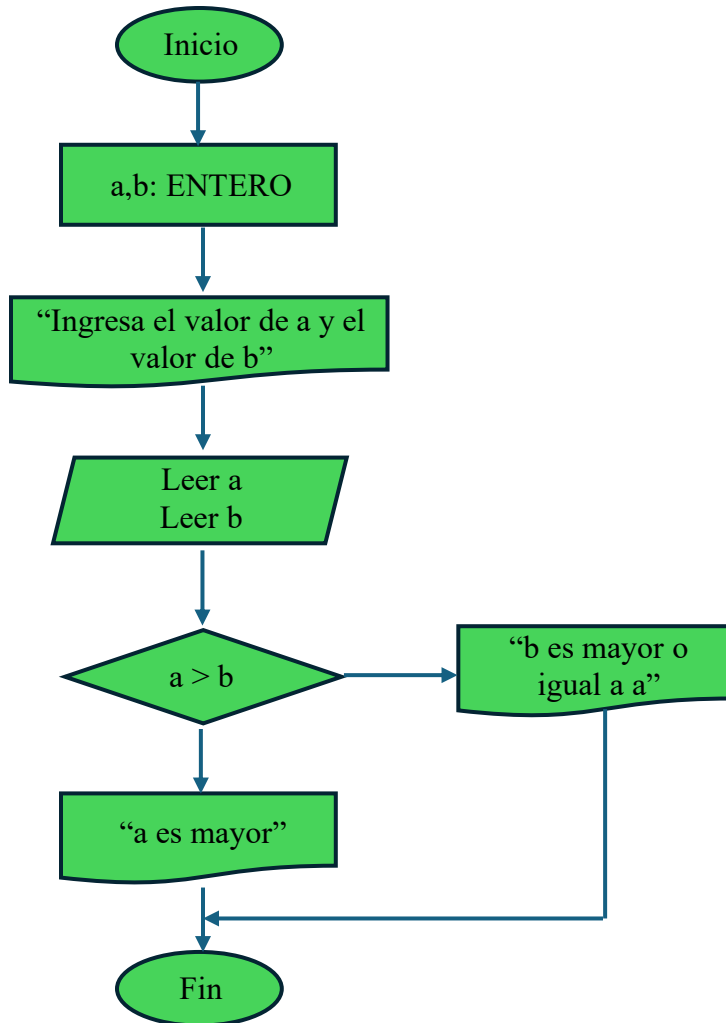


Figura 7. Ejemplo de estructura de control iterativa HACER-PARA (Tutoriales Programación Ya, s/f).

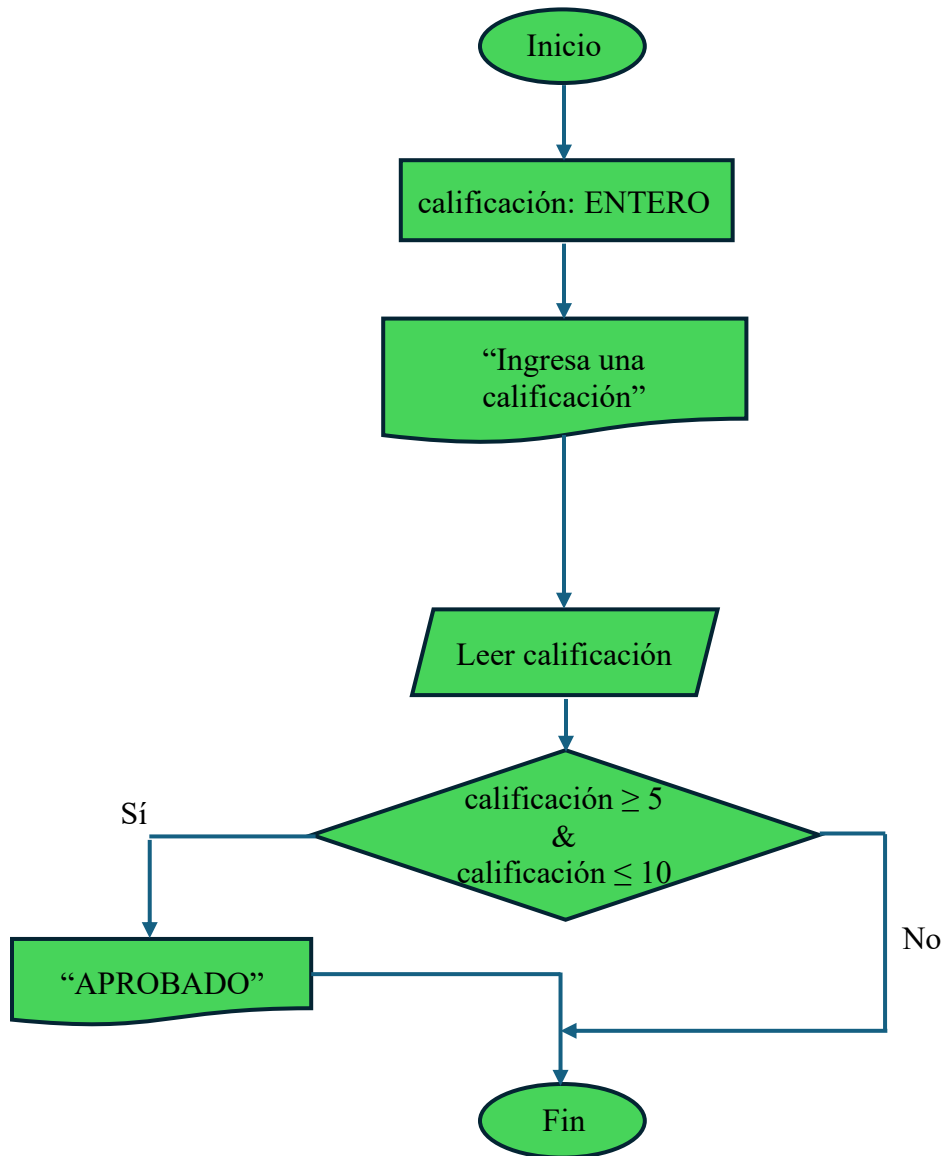
Ejercicio de clase:

1. Modificar la Figura 10 para que le solicite al usuario los valores de a y b.



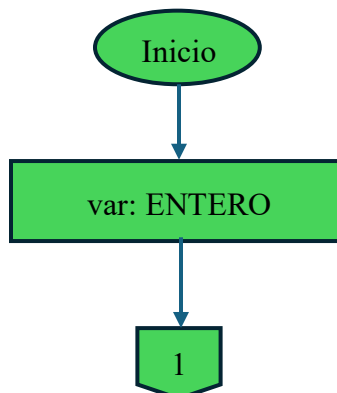
Iteración	a	b	Salida
1	2	5	b es mayor o igual a a
2	14	14	b es mayor o igual a a
3	49	37	a es mayor

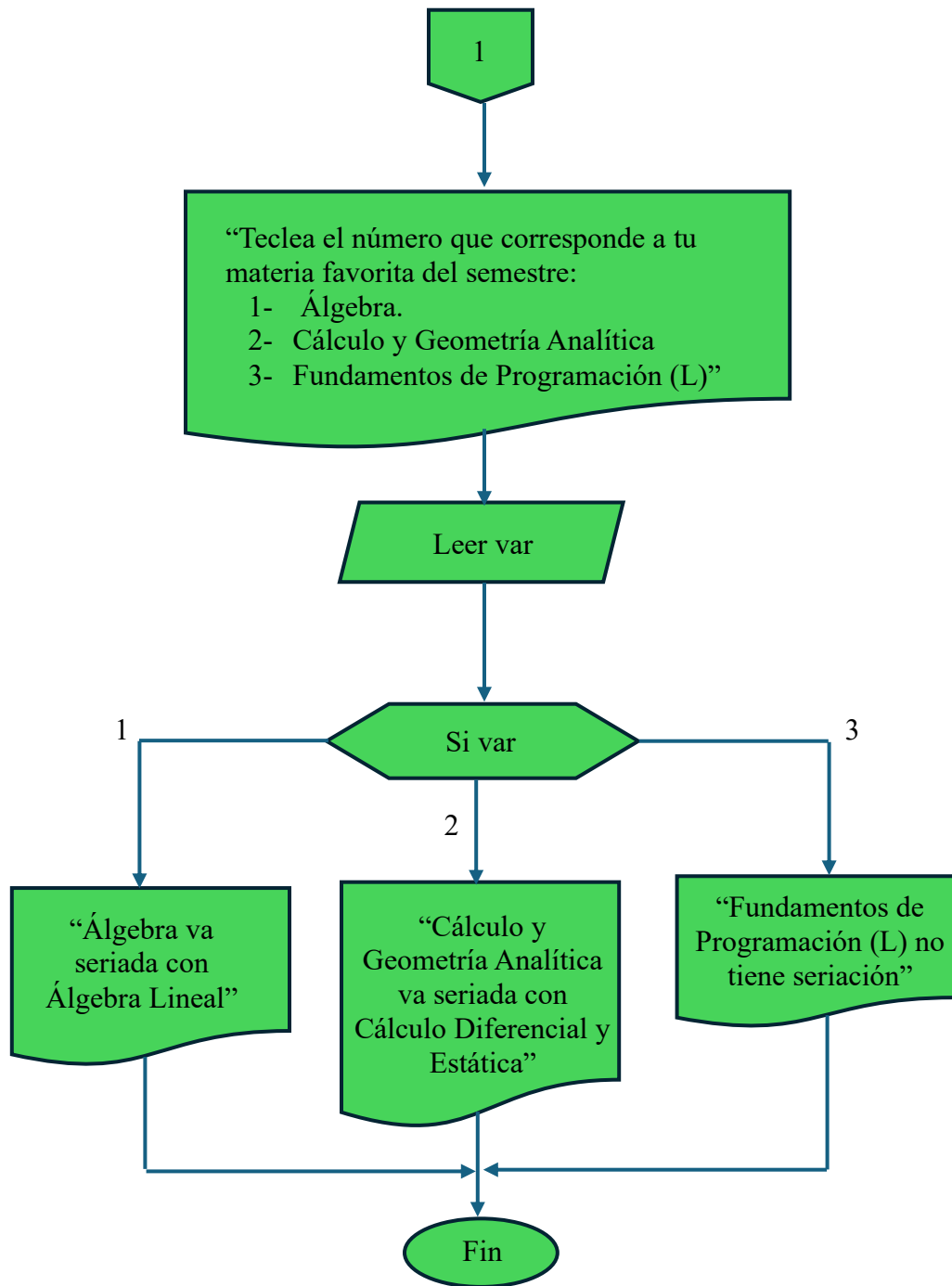
2. Solicita una calificación al alumno entre 5 y 10. En caso contrario ir a Fin, es decir, termina el programa.



Iteración	calificación	Salida
1	6	APROBADO
2	4	APROBADO
3	9	-

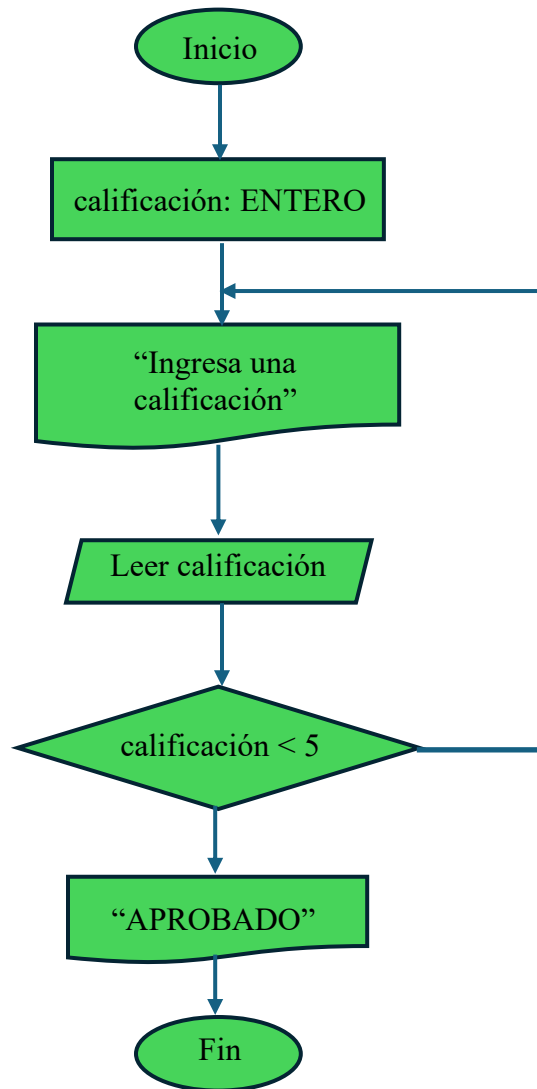
3. Desarrolla un menú libre de 3 opciones.





Iteración	var	Salida
1	3	Fundamentos de Programación (L) no tiene seriación.
2	1	Álgebra va seriada con Álgebra Lineal.
3	2	Cálculo y Geometría Analítica va seriada con Cálculo Diferencial y Estática.

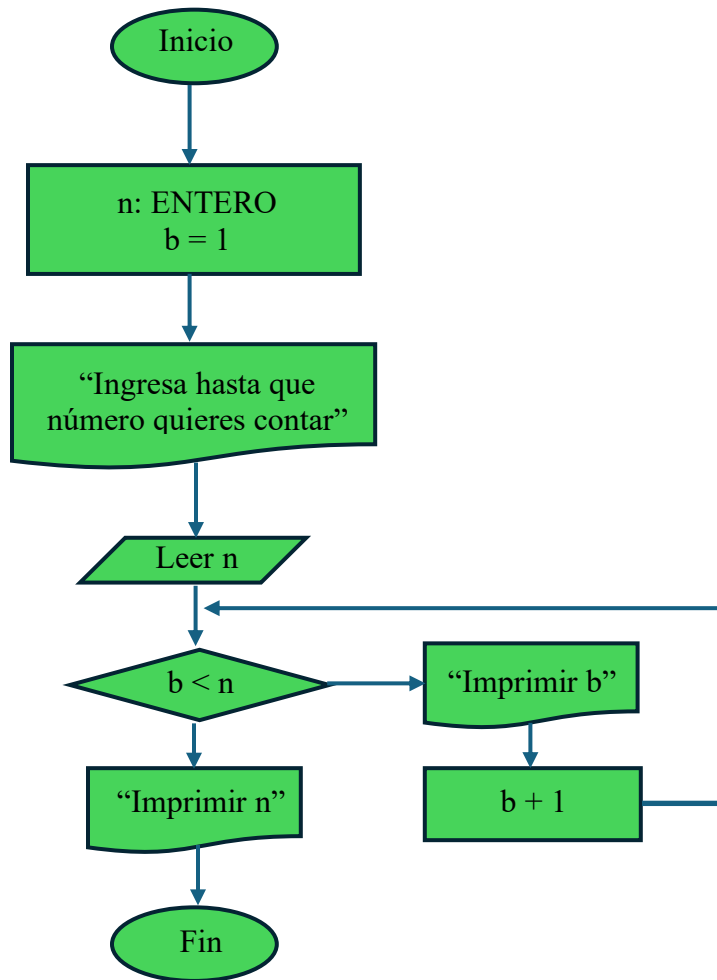
4. Solicitar una calificación entre 5 y 10. En caso de que no cumpla, volver a solicitar la calificación.



Iteración	var	Salida
1	7	APROBADO
2	2	-
3	4	-

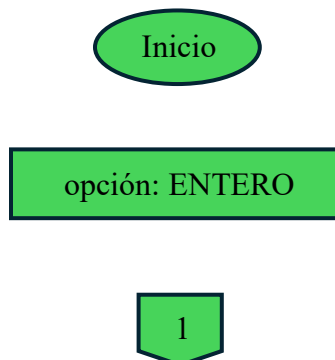
Ejercicio de tarea:

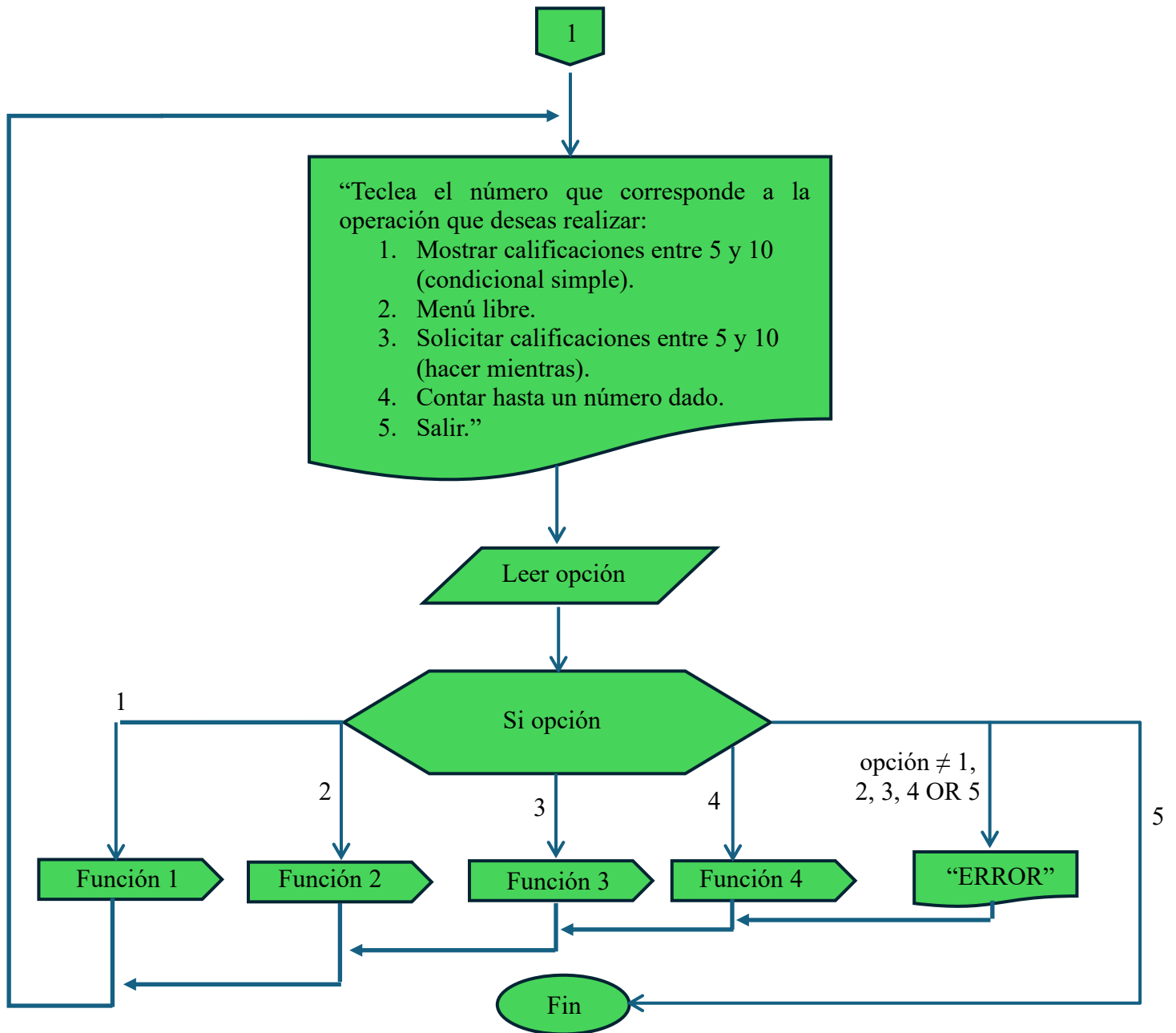
1. Solicita un número entero positivo y cuenta desde 1 hasta el número dado (estructura hacer mientras y mientras).



Iteración	n	Salida
1	10	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
2	15	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
3	20	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20

2. Genera un menú de los 4 programas (3 de clase y el de tarea), agregar el ciclo hacer mientras, en donde valide, si el usuario no elige las opciones 1, 2, 3 o 4 imprimir un letrero de "error" (es el default) y volver a mostrarle el menú, hasta que seleccione la opción 5 como salir.





Iteración	opción	Salida
1	1	APROBADO
2	4	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
3	5	-
4	3	APROBADO
5	7	ERROR
6	2	Álgebra va seriada con Álgebra Lineal.

Conclusiones:

Sin lugar a dudas, a partir de esta práctica considero que pude ver de manera mucho más ejemplificada mi avance personal y aprendizaje durante lo que va de la asignatura de Fundamentos de Programación. Hacer diagramas de flujo es todo un proceso en sí mismo que ayuda a saber o localizar errores en un algoritmo ya que permite establecer relaciones más visuales entre los pasos a seguir para la solución de un problema.

A nivel conceptual, saber “leer” o entender un diagrama de flujo puede no ser tan sencillo para algunas personas ya que implica tener un conocimiento previo sobre la simbología utilizada, así como del establecimiento previo de un algoritmo o problema y sus datos de entrada y/o salida.

Me parece que esta práctica fue muy bien llevada tanto en el laboratorio como de manera asincrónica como tarea. Sin duda se ve un buen trabajo docente por parte de la profesora al darnos la oportunidad de diseñar diagramas de flujo con la mayor cantidad de estructuras de control posibles, esto obvio con el fin de reforzar los conocimientos teóricos estudiados anteriormente.

Aunque para la fecha de entrega de esta práctica, ya hemos revisado un poco el tema de pseudocódigos, me parece que estoy listo para la práctica de dicho tema gracias a las actividades que se han realizado previamente tanto en el aula como en el laboratorio.

Fuentes de consulta:

- Facultad de Ingeniería. (2025). Manual de prácticas del laboratorio de Fundamentos de Programación. Laboratorio de computación. Salas A y B. Universidad Nacional Autónoma de México. pp. 52-72. Recuperado el 12 de septiembre de 2025 de <http://lcp02.fi-b.unam.mx/>