|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Karina García Morales. |
| *Asignatura:* | Fundamentos de programación |
| *Grupo:* | 21 |
| *No de Práctica(s):* | 03 |
| *Integrante(s):* | Vega Báez Erick Iván. |
|  |  |
|  |  |
| *Semestre:* | 2020-2 |
| *Fecha de entrega:* | 26-02-2020 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**Práctica 03: Solución de problemas y Algoritmos.**

**Objetivo**

Elaborar algoritmos correctos y eficientes en la solución de problemas siguiendo las etapas de Análisis y Diseño pertenecientes al Ciclo de vida del software.

**SOLUCIÓN A UN PROBLEMA.**

Un problema informático se puede definir como el conjunto de instancias al cual corresponde un conjunto de soluciones, junto con una relación que asocia para cada instancia del problema un subconjunto de soluciones.

Para poder solucionar un problema nos apoyamos en la Ingeniería de Software que de acuerdo a la IEEE se define como “La aplicación de un enfoque sistemático, disciplinado y cuantificable hacia el desarrollo, operación y mantenimiento del software". Por lo que el uso y establecimiento de principios de ingeniería sólidos, son básicos para obtener un software que sea económicamente fiable y funcione eficientemente.

La Ingeniería de Software provee métodos que indican cómo generar software. Estos métodos abarcan una amplia gama de tareas:

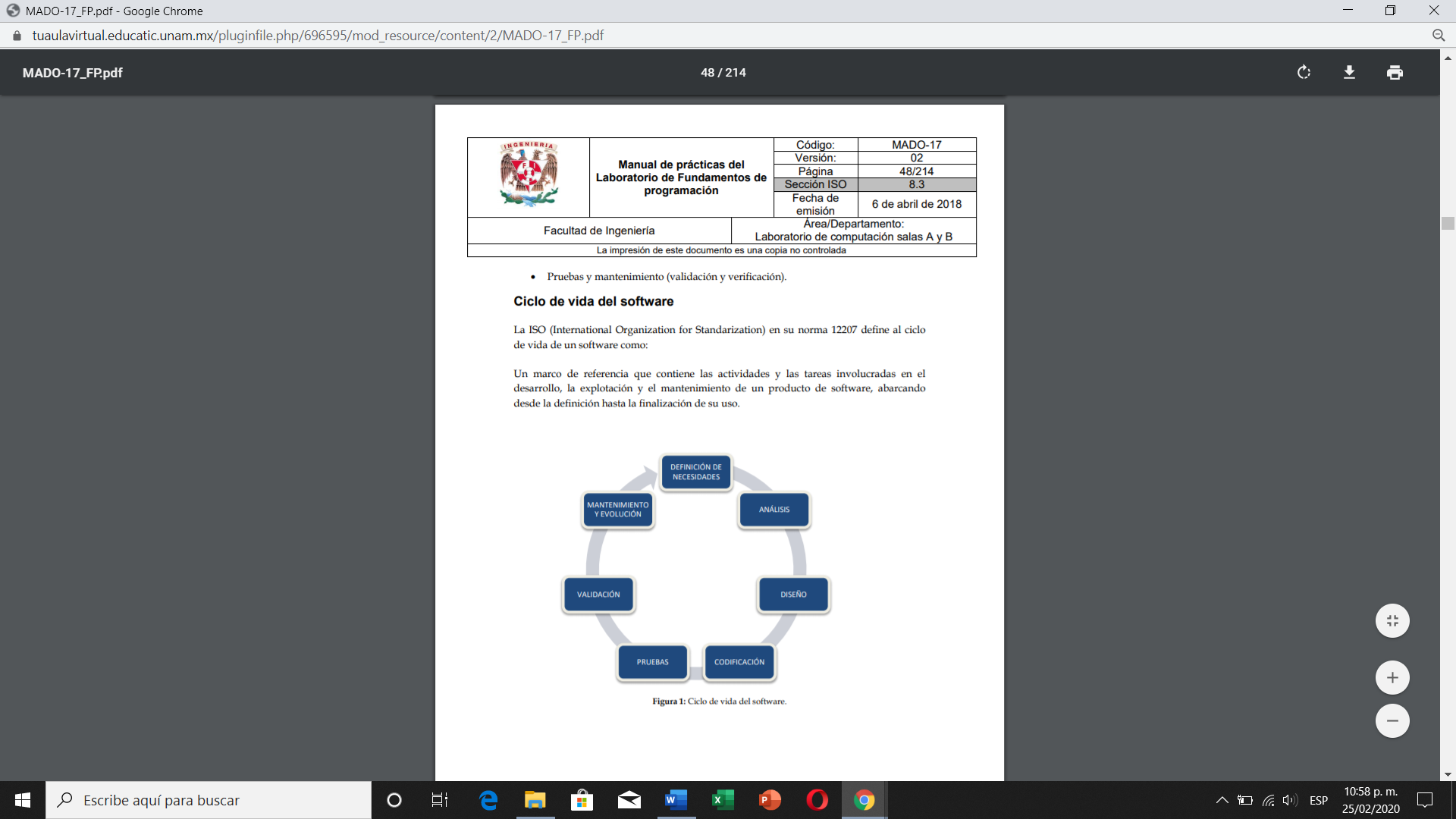
• Planeación y estimación del proyecto.

• Análisis de requerimientos del sistema y software.

• Diseño de la estructura de datos, la arquitectura del programa y el procedimiento algorítmico.

• Codificación

**CICLO DE VIDA DEL SOFTWARE.**



Definición de Necesidades: En esta fase se prepara el diseño y posterior implementación del sistema. Es necesario definir el alcance del proyecto, justificarlo y escoger una metodología para su desarrollo. También es preciso asociar las diferentes actividades a plazos de tiempo y designar roles y responsabilidades.

Análisis: **Es la que tiene que ver con el análisis y donde se busca concretar una serie de requisitos,** que son los que regirán el nuevo sistema o los cambios a introducir en el antiguo, si con el proyecto se busca su actualización.

Diseño: En este estadio el equipo de proyecto tendrá que determinar cómo el nuevo sistema de información cumplirá con los requisitos aplicables. Es por ello que**, a estas alturas del ciclo de vida de un sistema de información conviene identificar soluciones potenciales, evaluarlas y elegir la más conveniente**. Ésta será o la más efectiva, o la más eficiente en costes o la menos compleja

Codificación: El desarrollo software marca un antes y un después en la vida del sistema y significa, además, el inicio de la producción. El cambio es una constante durante esta etapa, en la que suele ser recomendable poner el foco en la formación y capacitación de los usuarios y el equipo técnico.

Pruebas: El objetivo de esta etapa es corroborar que el diseño propuesto cumple con los requisitos de negocio establecidos. Puede ser necesario repetir las pruebas tantas veces como haga falta para evitar errores y, de hecho, conviene que el usuario final dé su conformidad con el resultado.

Validación: **En esta etapa del ciclo de vida de un sistema de información hay que proceder a la instalación del hardware y software elegidos**, crear las aplicaciones correspondientes, someterlas a pruebas, crear la documentación pertinente y capacitar a los usuarios.

Mantenimiento y evolución: **Esta etapa del ciclo de vida de un sistema de información está relacionada con las operaciones del día a día. Por lo general, consiste en introducir los ajustes necesarios** para mejorar el rendimiento y corregir los problemas que puedan surgir.

**Datos de entrada:** Está compuesto por todos aquellos datos que pueden alimentar al sistema.

**Datos de Salida:** Está compuesto por todos los datos que el sistema regresará como resultado del proceso. Estos datos se obtienen a partir de los datos de entrada.

La unión del conjunto de entrada y el conjunto de salida forman lo que se conoce como el dominio del problema, es decir, los valores que el problema puede manejar.

**Construcción del algoritmo**: Sé refiere a la descripción detallada de los pasos que deben seguirse para resolver el problema.

**Desarrollo**

Determinación de signo.

Problema: Determinar si un número dado es positivo o negativo.

**Datos de entrada:** Número real.

**Datos de Salida:** La validación de si el número es positivo

**Datos Auxiliares:** El número no puede ser cero.

1. Inicio.
2. Solicitar un número Real.
3. Leer el número, sí el número proporcionado es 0, regresar al paso 1.

Sí el número ingresado es diferente a cero, validar lo siguiente.  
4.1 Sí el número es mayor a 0, su signo será “+”.

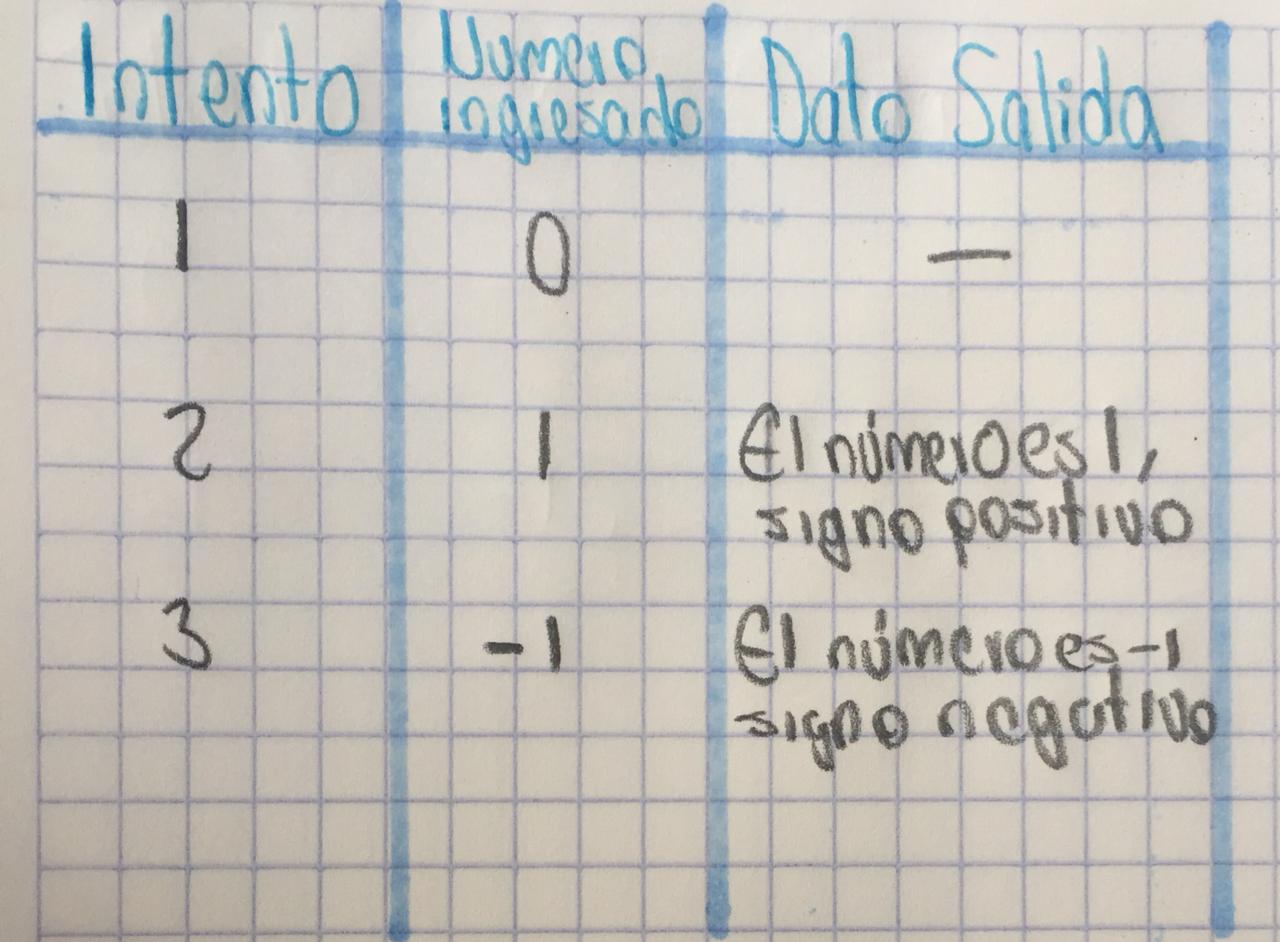
4.2 Sí el número es menor a 0, su signo será “-“.

1. Se establecen las condiciones, sí el resultado arrojado es “+”, arrojar como resultado “0”

4.1 Si el resultado es “-“, arrojar como resultado 1.

1. Imprimir signo.
2. Fin

PRUEBAS DE ESCRITORIO.



Programa Factorial.

Problema: Resolver factorial de números entre 1 y 10

**Datos de entrada:** Número entero.

**Datos de Salida:** Impresión de la factorial del número ingresado.

**Datos Auxiliares:** El número debe ser >0, pero <11

1. Inicio.
2. Declarar contador ( cont) en cero, factorial (fact) 1
3. Pedir número del que se desea conocer el factorial (m)
4. Se establecen las condiciones Si 0 < m < 11 ir al paso 4; en caso contrario ir al paso 2
5. Mientras contador menor igual que m:

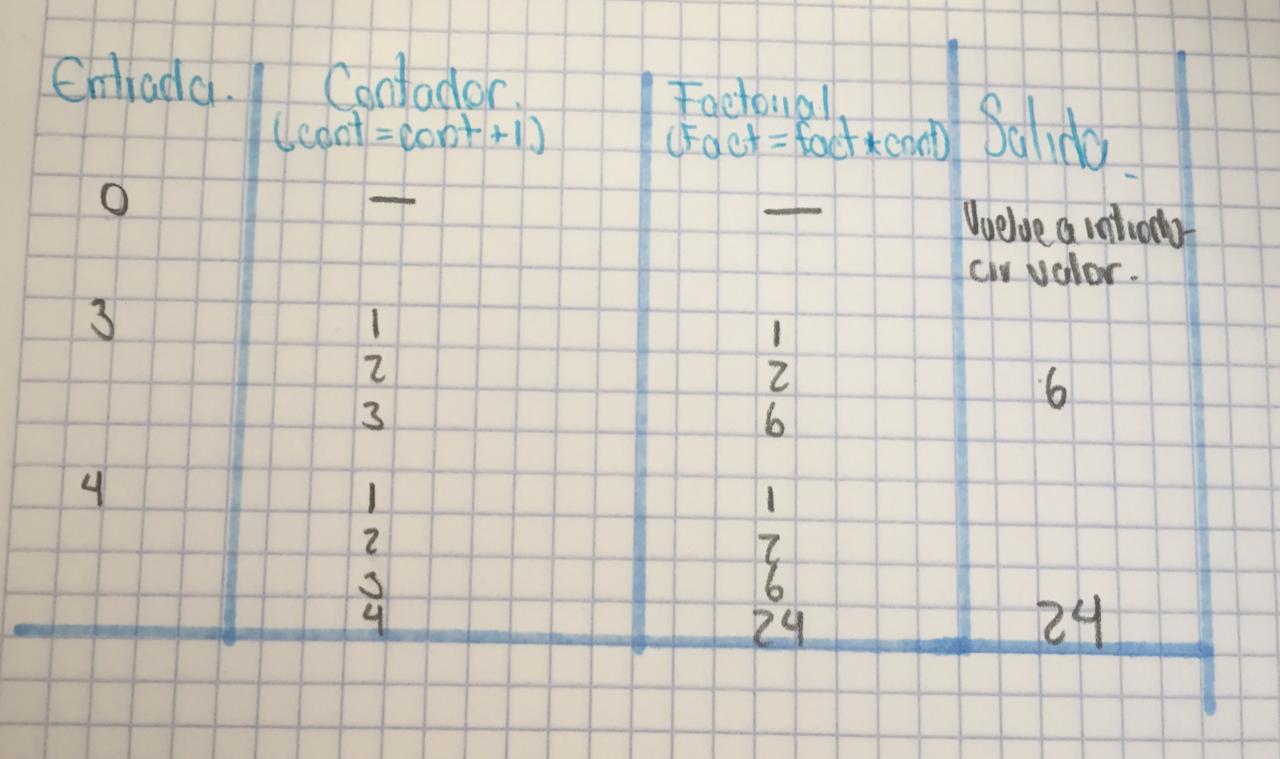
fact = fact \* cont

cont = cont + 1

en caso contrario ir al paso 7

1. Imprimir fact
2. Fin

PRUEBA DE ESCRITORIO



Obtención de una figura

Problema: Seguir el algoritmo para obtener una figura

**Datos de Entrada:** Hoja tamaño carta en limpio, regla y lápiz.

**Datos de Salida:** Figura correcta.

1. Dibuja una V invertida. Empieza desde el lado izquierdo, sube, y baja hacia el lado derecho, no levantes el lápiz.

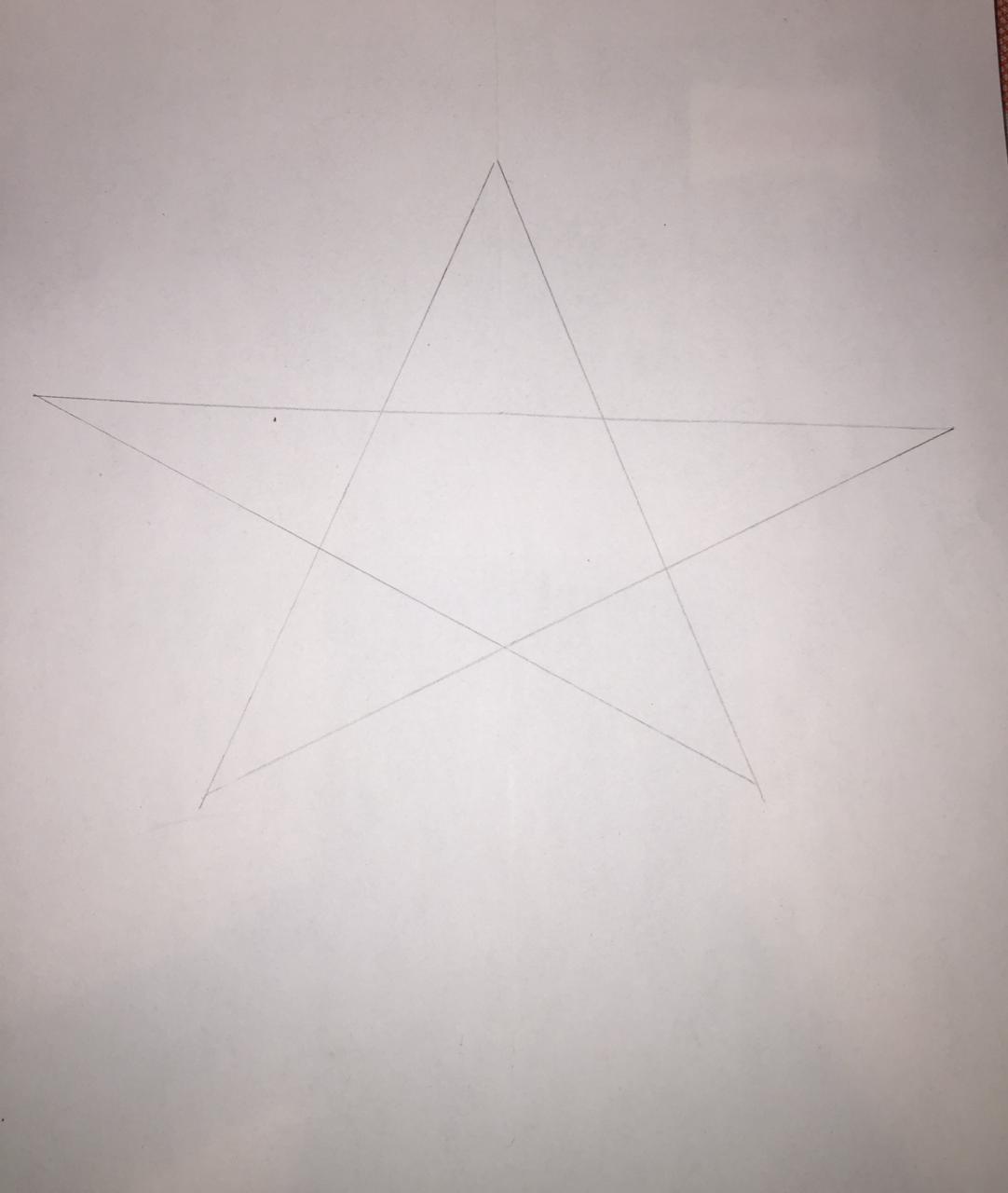
2. Ahora dibuja una línea en ángulo ascendente hacia la izquierda. Debe cruzar la primera línea más o menos a 1/3 de la altura. Todavía no levantes el lápiz del papel.

3. Ahora, dibuja una línea horizontal hacia la derecha. Debe cruzar la V invertida más o menos a 2/3 de la altura total. Sigue sin levantar el lápiz.

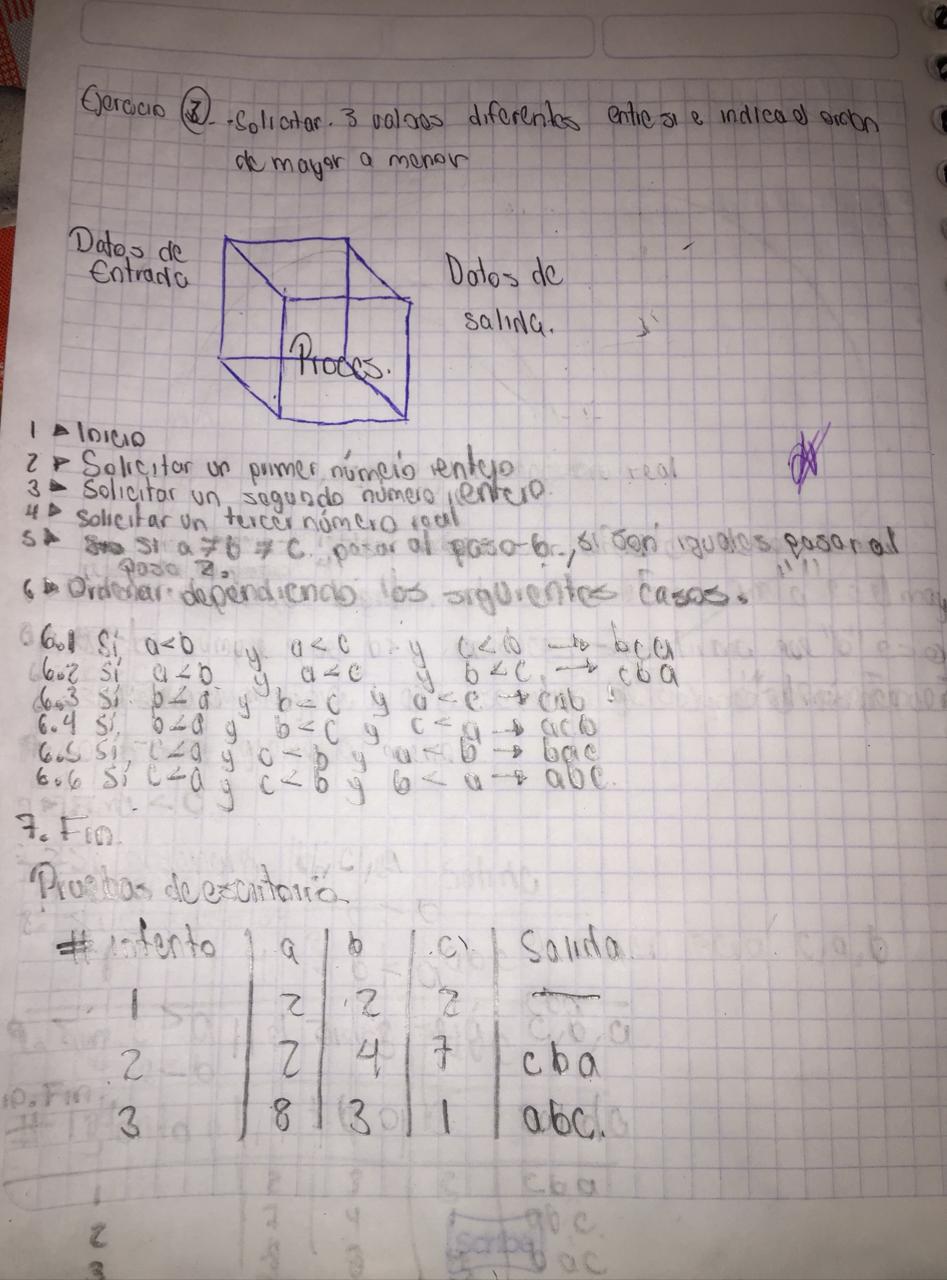
4. Dibuja una línea en un ángulo descendente hasta el punto de inicio. Las líneas deben unirse.

5. Ahora ya puedes levantar el lápiz del papel. Has terminado la estrella de 5 puntas.

PRUEBA DE ESCRITORIO

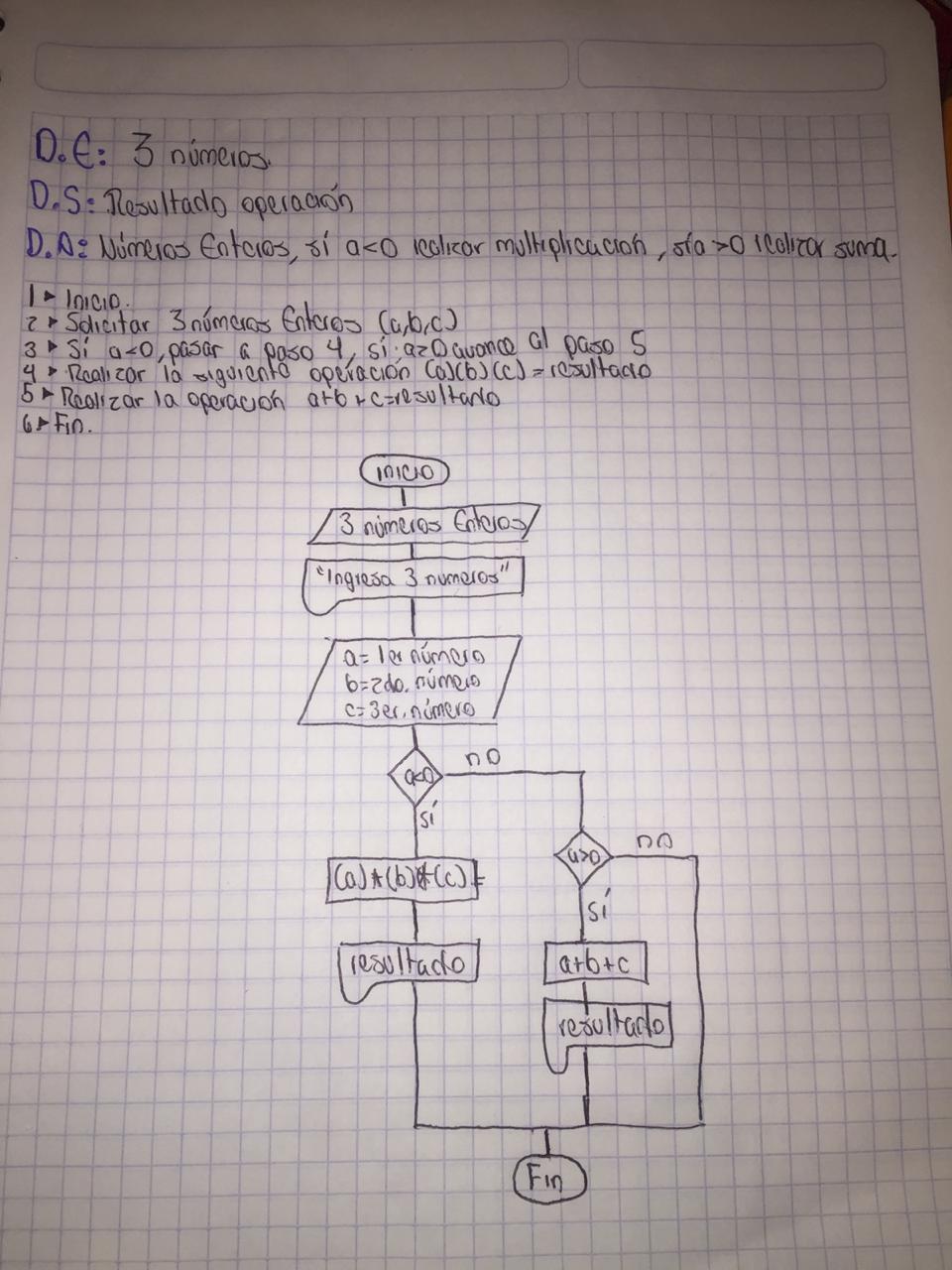


ACTIVIDAD 4.

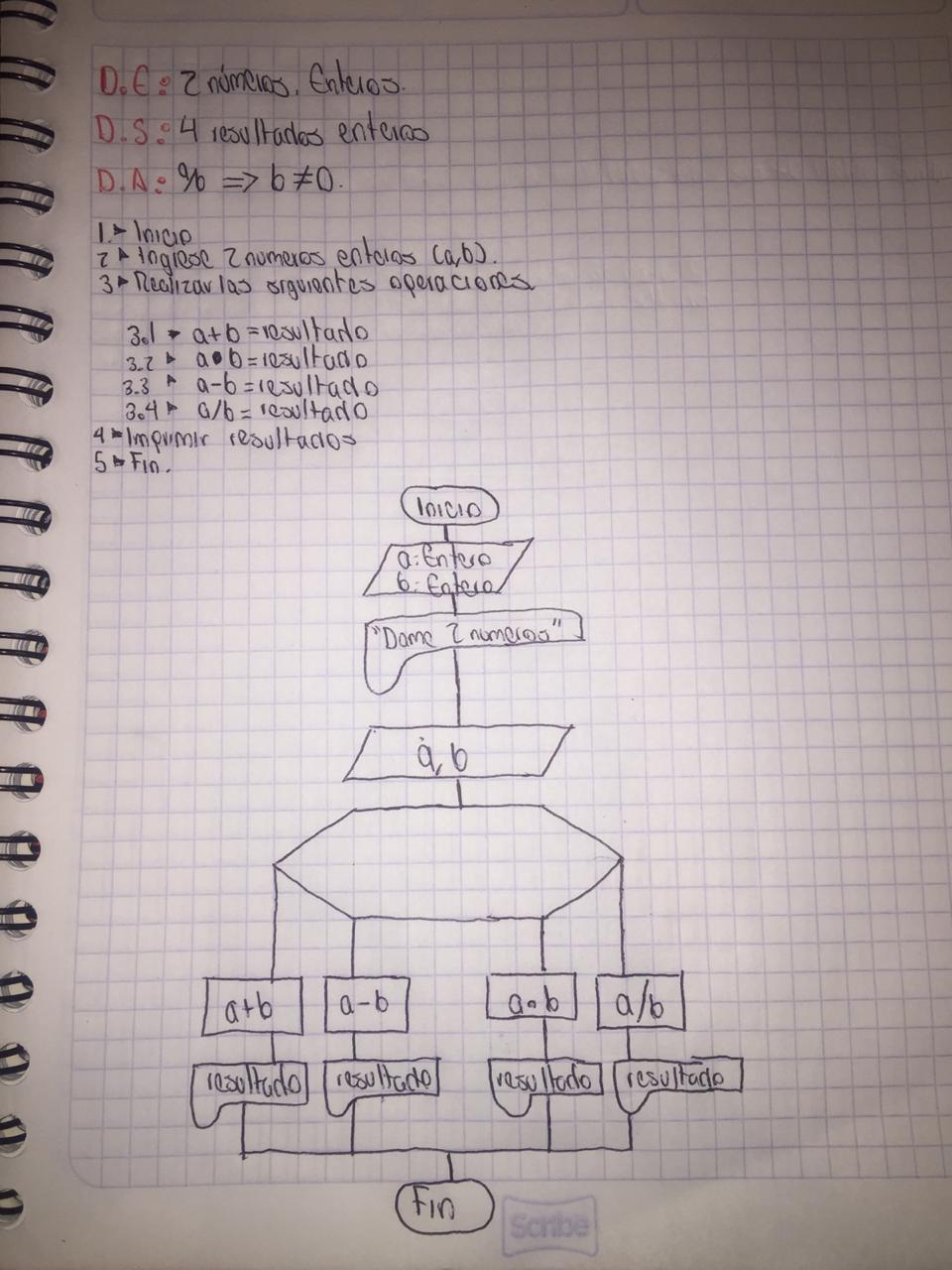


**TAREA.**

* Dados tres números; si el primero es negativo, deberá realizar el producto de los tres, y en caso contrario realizará la suma.



* Dados dos números realizar suma, resta, producto y división (considerar el dividendo ≠ 0).

****

Bibliografía.

<https://blog.powerdata.es/el-valor-de-la-gestion-de-datos/ciclo-de-vida-de-un-sistema-de-informacion-fases-y-componentes>

<https://tuaulavirtual.educatic.unam.mx/pluginfile.php/696595/mod_resource/content/2/MADO-17_FP.pdf>