



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* M.I. MARCO ANTONIO MARTÍNEZ QUINTANA

*Asignatura:* FUNDAMENTOS EN COMPUTACIÓN

*Grupo:* 1103

*No de Práctica(s):* #3

*Integrante(s):* RAUDALES PALMA LEONARDO DE JESÚS

*No. de Equipo de  
cómputo empleado:* NO APLICA

*No. de Lista o Brigada:*

*Semestre:* PRIMER SEMESTRE

*Fecha de entrega:* JUEVES 22 DE OCTUBRE

*Observaciones:*

**CALIFICACIÓN:** \_\_\_\_\_

## 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.

## ACTIVIDADES:

- A partir del enunciado de un problema, identificar el conjunto de entrada y el conjunto de salida.
- Elaborar un algoritmo que resuelva un problema determinado (dado por el profesor), identificando los módulos de entrada, de procesamiento y de salida.

## INTRODUCCIÓN:

### Algoritmos

Una vez realizado el análisis, es decir, ya que se entendió qué es lo que está solicitando el usuario y ya identificado el conjunto de entrada y el conjunto de salida, se puede proceder al diseño de la solución, esto es, a la generación del algoritmo.

Dentro del ciclo de vida del software, la creación de un algoritmo se encuentra en la etapa de diseño. Ver figura 1.

Durante el diseño se busca proponer una o varias alternativas viables para dar solución al problema y con base en esto tomar la mejor decisión para iniciar la construcción.

Un problema matemático es computable si éste puede ser resuelto, en principio, por un dispositivo computacional.

La teoría de la computabilidad es la parte de la computación que estudia los problemas de decisión que pueden ser resueltos con un algoritmo.

Un algoritmo se define como un conjunto de reglas, expresadas en un lenguaje específico, para realizar alguna tarea en general, es decir, un conjunto de pasos, procedimientos o acciones que permiten alcanzar un resultado o resolver un problema. Estas reglas o pasos pueden ser aplicados un número ilimitado de veces sobre una situación particular.

Un algoritmo es la parte más importante y durable de las ciencias de la computación debido a que éste puede ser creado de manera independiente tanto del lenguaje como de las características físicas del equipo que lo va a ejecutar.

Las principales características con las que debe cumplir un algoritmo son:

- Preciso: Debe indicar el orden de realización de paso y no puede tener ambigüedad
- Definido: Si se sigue dos veces o más se obtiene el mismo resultado.
- Finito: Tiene fin, es decir tiene un número determinado de pasos.
- Correcto: Cumplir con el objetivo.
- Debe tener al menos una salida y esta debe de ser perceptible
- Debe ser sencillo y legible
- Eficiente: Realizarlo en el menor tiempo posible
- Eficaz: Que produzca el efecto esperado

## DESARROLLO:

### *Ejercicio 1*

**PROBLEMA:** Seguir el algoritmo para obtener una figura

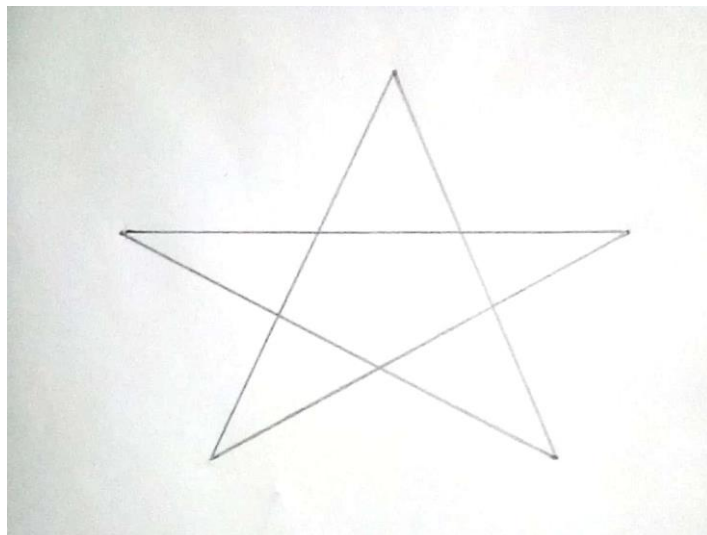
**ENTRADA:** Hoja tamaño carta en limpio, regla y lápiz.

**SALIDA:** Figura correcta.

### Algoritmo

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.

Realizando los pasos del algoritmo planteado, obtenemos la siguiente figura:



## Ejercicio 2

**PROBLEMA:** Seguir el algoritmo para obtener una figura

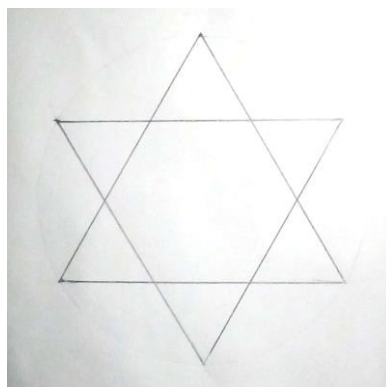
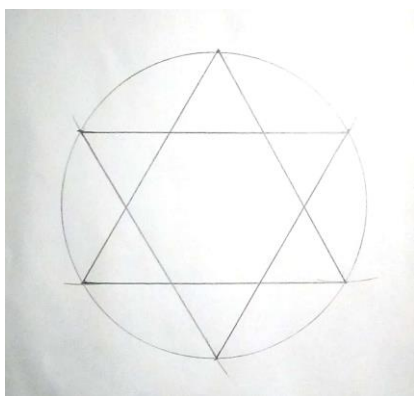
**ENTRADA:** Hoja tamaño carta en limpio, regla y lápiz.

**SALIDA:** Figura correcta.

### Algoritmo

1. Empieza dibujando un círculo con un compás. Coloca un lápiz en el compás. Coloca la punta del compás en el centro de una hoja de papel.
2. Ahora gira el compás, mientras mantienes la punta apoyada en el papel. El lápiz dibujará un círculo perfecto alrededor de la punta del compás.
3. Marca un punto en la parte superior del círculo con el lápiz. Ahora, coloca la punta del compás en la marca. No cambies el radio del compás con que hiciste el círculo.
4. Gira el compás para hacer una marca en el propio círculo hacia la izquierda. Haz una marca también en el lado derecho.
5. Ahora, coloca la punta del compás en uno de los puntos. Recuerda no cambiar el radio del compás. Haz otra marca en el círculo.
6. Continúa moviendo la punta del compás a las otras marcas, y continúa hasta que tengas 6 marcas a la misma distancia unas de otras. Ahora, ya puedes dejar tu compás a un lado.
7. Usa una regla para crear un triángulo que empiece en la marca superior del círculo. Coloca el lápiz en la marca superior. Ahora dibuja una línea hasta la segunda marca por la izquierda. Dibuja otra línea, ahora hacia la derecha, saltándote la marca de la parte más baja. Complementa el triángulo con una línea hacia la marca superior. Así completarás el triángulo.

Siguiendo los pasos del algoritmo podemos realizar la figura de la estrella de 6 picos, borrando el círculo que hicimos de referencia se ve claramente la figura.



## CONCLUSIONES:

Se puede concluir que los algoritmos son una parte esencial en el ciclo de vida del software, pues brindan la oportunidad de crear un orden de pasos a seguir para lograr resolver el problema planteado. Este algoritmo puede ser similar a un instructivo algo riguroso, porque nos indica en regla bastante inflexible los pasos que vamos a realizar, es preciso y lógico, a la vez que debe ser comprensible para cualquier persona, estas características hacen que el algoritmo nos permita llegar a la respuesta que esperamos para poder resolver el problema. Son importantes los datos de entrada y los datos de salida pues nos dan los datos indispensables sobre el análisis del problema, no podríamos saber por qué cada cosa que realizamos si no fuera por estos datos. Así como las restricciones, si no se plantearan límites el proyecto podría tender a ser muy ambiguo, porque la mar de posibilidades es inmensa, reduciendo esas posibilidades cada vez llegas a un margen más pequeño y justo de lo que realmente es la necesidad, gracias a estas restricciones puedes ser capaz de encontrar una aguja en un pajar.

El ciclo de vida de un software para mí es la construcción base de todo proyecto computacional, cada paso tiene que ser cuidado meticulosamente para poder obtener un producto funcional, es por eso que, para cualquier idea, debe realizarse.