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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | RODRIGUEZ ESPINO CLAUDIA |
| *Asignatura:* | FUNDAMENTOS DE PROGRAMACION |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | 3 |
| *Integrante(s):* | SERRANO HERNANDEZ ITZEL ALEJANDRA |
|  |  |
| *Semestre:* | PRIMERO |
| *Fecha de entrega:* | 03-09-2017 |
| *Obervaciones:* |  |
|  |  |

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.

En esta ocasión como de costumbre leímos la práctica, aclarando con la profesora algunas dudas acerca de ella. Esta se basó en algoritmos y sus componentes, así como en algunos ejemplos.

Cuando finalizamos las dudas la profesora dio indicaciones de que hiciéramos tres “programas” diferentes para reforzar lo aprendido; así como, un ejemplo de prueba de escritorio que venía en la práctica.

Los programas y el ejemplo se muestran a continuación:

**PROG. 1**

FORMULA GENERAL

Datos de entrada:

Datos de salida: Valor de X1 y X2

Algoritmo:

1. Inicio.
2. Pedir un valor para “a”. Mostrar mensaje “No debe ser 0”
3. Si a=0 Mostrar “Crea indeterminación. Intenta de nuevo” y regresar al paso 2, en caso contrario continuar en el paso 4.
4. Pedir valor para “b” y “c”
5. Realizar la operación b2 -4ac=d
6. Si d<0 entonces realizar operación =Z
7. Hacer la operación r=-b/2a, n=Z/2a y mostrar x1 = r+ni y x2 =r-ni
8. x1= , x2=
9. Mostrar x1 y x2
10. Fin.

PRUEBA DE ESCRITORIO

Raíz negativa.

1. a= 2
2. b= 4 c=6
3. (4)2 – 4(2)(6) =-32
4. =
5. =x1 ,  =x2
6. x1= 0.4142 x2=-2.4142

Raíz positiva.

1. a=2
2. b=4 c=1
3. (4)2-4(2)(1) =8
4. =x1 ,  =x2
5. x1= -.2929 x2=-1.7071

**PROG. 2**

FACTORIAL

Datos de entrada:

* Números del 1 al 9
* Debemos tomar en cuenta los números naturales

Datos de salida:

Resultado de la operación.

Algoritmo:

1. Inicio.
2. Contador=1
3. Pedir numero: m, Mostrar “Debe ser de 1-0”
4. Si 0< m <11 ir al paso 4, en caso contrario regresar al paso 3
5. Mientras contador ≤ 0 < m, realiza:

Fact=m \*(contador)

contador=m+1

1. Detener cuando contador +1 sea m+1, Resultado= F
2. Mostrar F
3. Fin.

Prueba de escritorio.

1.

1. m=4
2. Fact= 1\*2\*3\*4
3. Resultado: 24

2.

1. m=5
2. Fact= 1\*2\*3\*4\*5
3. Resultado: 120

**PROG 3.**

FORMULAS

Datos de entrada:

* Si y <0
* X1= y2 +4y-25

Datos de salida:

Valor de X1

Algoritmo:

1. Inicio.
2. Pedir valor de “y”, mostrar “ Tiene que ser menor a 0”
3. Si y ≥ 0 mostrar “y tiene que ser menor a 0, intenta de nuevo”
4. Si y < 0 ir al paso 6
5. Si y ≥ 0 regresar al paso 1
6. Multiplicar: y\*y=a
7. Multiplicar: 4\*y=b
8. Realizar operación a+b-25=X1
9. Mostrar “c”
10. Fin.

Prueba de escritorio.

1. y=-4
2. (-4) (-4) =16
3. (4) (-4) =-16
4. 16+(-16)-25=-25
5. -25= X1

2.

Datos de entrada:

* Si y > 2
* X1= 4y2-3y+0

Algoritmo:

1. Inicio.
2. Pedir valor de “y”, mostrar “Debe ser mayor a 2”
3. Si y ≤ 2 mostrar “Y debe ser mayor a 2”
4. Si y ≤ 2 regresar al paso 1
5. Si y >2 ir al paso 6
6. Multiplicar: y\*y\*4=a
7. Multiplicar: y\*3=b
8. Resolver operación: a-b+0= X1
9. Mostrar X1
10. Fin.

Prueba de escritorio.

1. y=4
2. (6)(6)(4) =144
3. (6)(3) = 18
4. 144-18+0=X1
5. X1=126

Ejemplo 1

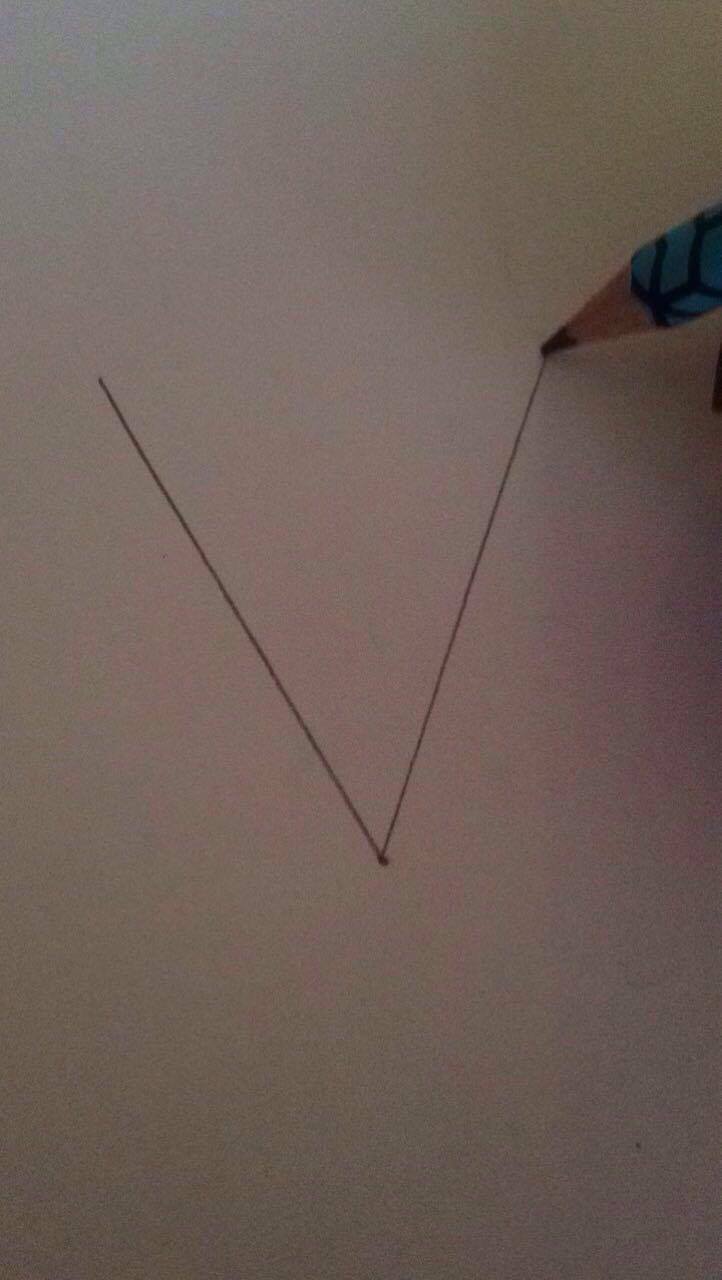
PROBLEMA: Seguir el algoritmo para obtener una figura

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

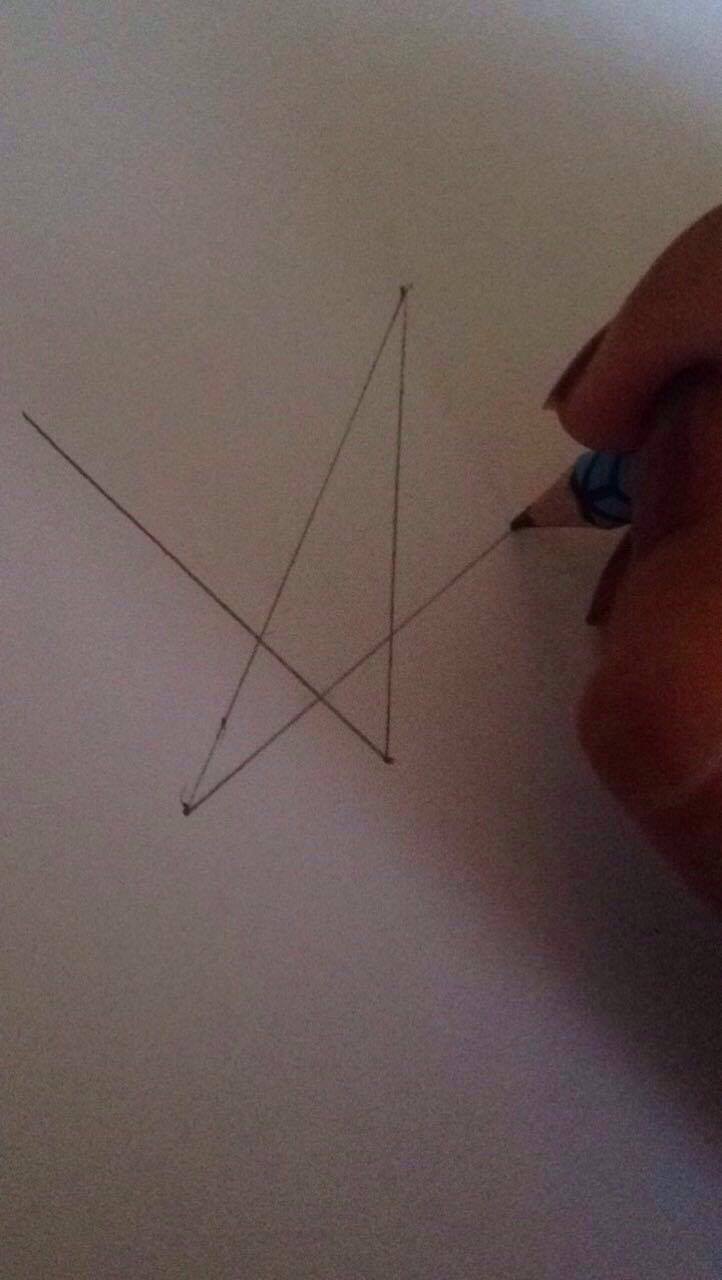
SALIDA: Figura correcta.

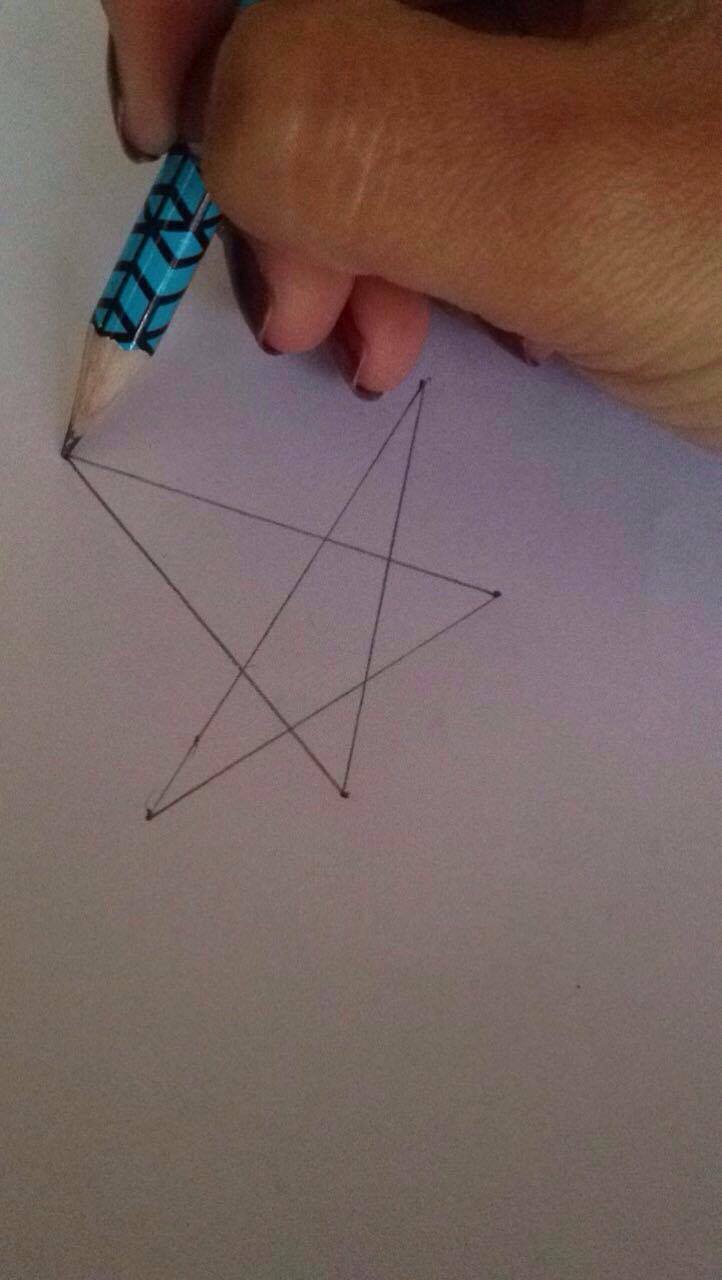
Prueba de escritorio:

Paso 1:



Paso 2 y 3:



Paso 4 y 5:

**CONCLUSONES:**

Los algoritmos en general son sumamente utilizados en nuestra vida diaria, formularlos son la base para resolver un problema.

Aprender sus componentes e identificarlos me ayudo a comprender más fácilmente algunos ejercicios que ya hacíamos hecho con anterioridad.

Esta práctica nos ayudará a enfocaros en ellos no solo en la materia, también lo hará en otras áreas de nuestra vida académica.