|  |  |  |
| --- | --- | --- |
|  | 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 programación |
| *Grupo:* | 02 |
| *No de Práctica(s):* | Práctica #3 |
| *Integrante(s):* | Martínez Martínez Yanni |
|  |  |
|  |  |
| *Semestre:* | 2018-1 |
| *Fecha de entrega:* | 1/Septiembre/2017 |
| *Obervaciones:* |  |
|  |  |

**Práctica 3: Solución de problemas y Algoritmos.**

**Objetivos:**

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

**Desarrollo:**

La práctica se leyó en el salón y de manera grupal, con ello se reconoció el ciclo de vida del software, y sus componentes correspondientes. Se comprendió más sobre un conjunto de entrada y uno de salida, así como el análisis de un algoritmo y como debe plantearse, sus pasos a seguir, su elaboración, y módulos básicos:

* Módulo de entrada, de procesamiento
* Módulo de salida
* Prueba de escritorio.

Para después comenzar a realizar la elaboración de practica de 4 programas los cuales eran; Ecuación de 2do grado, Factorial, 2 ecuaciones y 1 presente en el documento guía.

**Programas**

**Programa 1:** Ecuación de segundo grado

**Problema:** Resolver la fórmula general con los datos a, b, c que ingrese el usuario y lanzar los dos posibles resultados (suma y resta)

**Restricciones:** La variable *“a”* no puede ser cero. Las raíces no deberían ser negativas porque no son posibles resolverlas (imaginarias)

**Datos de entrada**: variables a, b, c

**Datos de salida:** La impresión de

**Dominio:** Todos los números reales

**Fórmula:**

**Algoritmo:**

1. Inicio

2. Solicitar un valor para *"a"*

3. Si *“a”* es igual a 0, desplegar el mensaje: "No es posible que sea cero"

3.1. Si *“a”* es igual a cero regresar al paso 2

3.2. Si *“a”* no es igual a cero continuar al paso 4

4. Pedir valores para *“b”* y *“c”*

5. Realizar operación:

5.1. Si *“d”* es menor a cero seguir al paso 6

5.2. Si *“d”* es mayor a cero ir a paso 7

6. Si *“d”* es menor a cero realizar la operación

6.1. Realizar operación y

6.2. Imprimir y

7. Realizar operación y

7.1. Imprimir y

8. Fin

**Prueba de escritorio:**

a=2 Paso 4: (3)(3) - 4(2)(1)=1 d=1

b=3 paso 7: = y

c=1

d=1

**Programa 2:** Factorial

**Problema:** Obtener el factorial de un número mayor o igual a cero y menor o igual a once.

**Restricciones:** No puede ser negativo

**Datos de entrada:** Números positivos

**Datos de salida:** La impresión del factorial del número que ingresó el usuario

**Dominio:** Todos los números naturales

*Factorial mayor a cero y menor a 11*

**Algoritmo:**

1.Inicio

2.Pedir valor para *“m”*

3.validar

3.1 Si no se cumple esta regla ir al paso 2

3.2 Si se cumple esta regla ir al paso 4

4. Declarar contador=(contador+1) ; Declarar factorial=1

5. Realizar la fórmula

6. Regresar al paso 4 y repetir hasta que contador sea igual a m

7. Impresión del factorial

**Prueba de escritorio:**

4!

m=4 mayor a 0 y menor que 11

**Programa 3:** Ecuaciones

**Problema:** Resolver dos tipos de ecuaciones, dependiendo del valor que ingrese el usuario resolver una ecuación en específico y así encontrar el valor de x

**Restricciones:** la variable *“y”* no puede ser igual a cero

**Datos de entrada:** Número cualquiera

**Datos de salida:** La impresión del valor de x

**Dominio:** Todos los números reales

**Ecuaciones:**

1. *si*
2. si

**Algoritmo:**

0. Inicio

1. Desplegar el mensaje de bienvenida: *"Hola. resuelve dos tipos de ecuación"*

2. Pedir valor para *“y”*

2.1. Si *“y”* es menor a dos ir al paso 3

2.2. Si *“y”* es mayor a dos ir al paso 4

2.3. Si *“y”* es igual a dos desplegar un mensaje: *"El numero dos no es un número válido, ingrese un número distinto"* y regresar al paso 2

3. Realizar la ecuación con el número que el usuario ingresó:

3.1.

3.2.

3.3. Realizar la operación y guardarla en variable

3.4 Ir al paso 5  
4. Realizar la ecuación con el número que el usuario ingresó:

4.1

4.2

4.3

4.4

5. Imprimir resultado

6. Fin

**Prueba de escritorio:**

**y=1**

**y=5**

**Programa 4:** Ejercicio 1 (Guía de práctica)

**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.

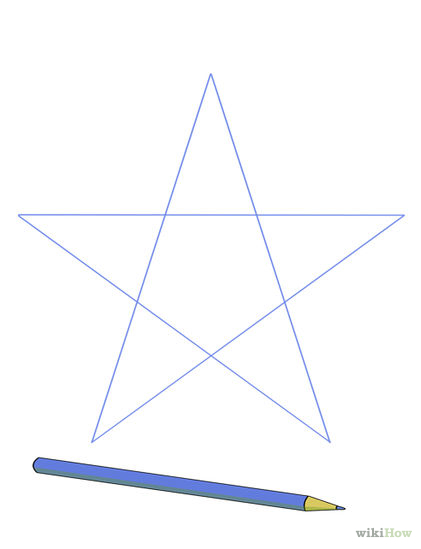
2. Ahora dibuja una línea en ángulo ascendente hacia la izquierda. Debe cruzar la primera línea más o menos a 1/3 de la altura. Todavía no levantes el lápiz del papel.

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.

**Prueba de escritorio:**



**Conclusiones:**

Con la ayuda de esta práctica pude aprender más acerca de una buena estructuración para un programa, es decir, el cómo realizar una buena estructura de algoritmos, observar de mejor manera su planteamiento de problema, sin olvidar que es importantísimo realizar pruebas de escritorio para ver que realmente funciona el programa, considero que esta práctica fue sumamente importante para una mejor comprensión al respecto, para que dichos conocimientos teórico-prácticos tengan frutos en el futuro de nuevas prácticas y con ello problemas más complejos .

**Bibliografía:**

* <http://lcp02.fi-b.unam.mx/>