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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Rodríguez Espino Claudia |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 03 |
| *No de Práctica(s):* | 3 Solución de Problemas y Algoritmos |
| *Integrante(s):* | JIMÉNEZ BARRANCO SOFIA |
|  |  |
|  |  |
| *Semestre:* | 2018-2 |
| *Fecha de entrega:* |  |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

***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.

**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**

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 (posiblemente vacío). 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.

• Pruebas y mantenimiento (validación y verificación).

**FÓRMULA GENERAL**

Algoritmo:

1. Inicio

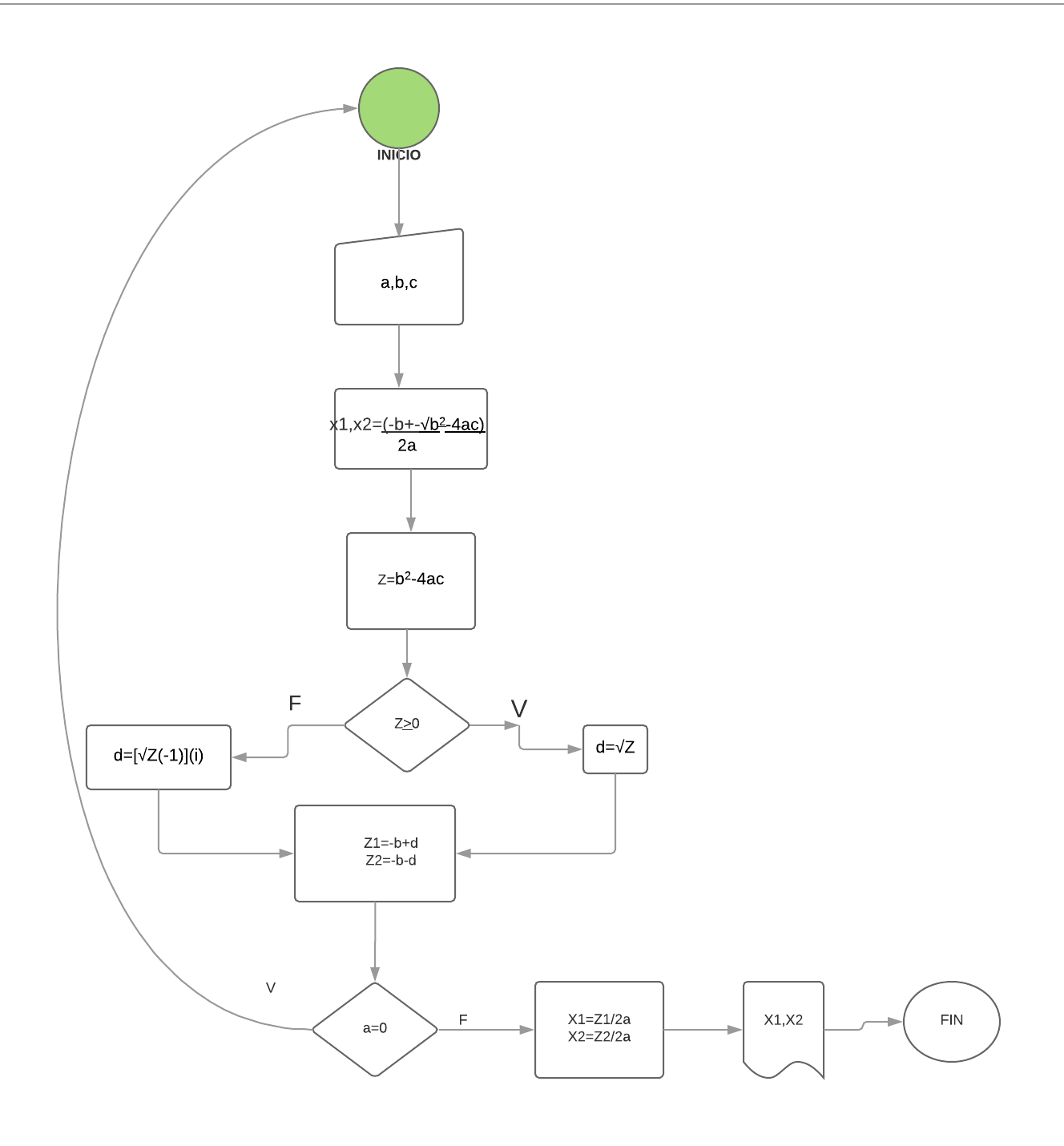
DATOS DE ENTRADA: 3 números reales

DATOS DE SALIDA: Dos resultados X1 y X2

DOMINIO: Números Reales e Imaginarios

Restricciones: a=0

1. Introducir datos de entrada a, b, c
2. Proceso x1,x2=(-b+-√b2-4ac) /2a
3. Z= b2-4ac
4. Si Z> 0 obtener la raíz que llamaremos “d”
5. Si Z<0 d=[√Z(-1)](i)
6. Z1= -b +d
7. Z2= -b –d
8. Si a=0 regresamos al inicio, de lo contrario continuar
9. Operamos X1=Z1/2a y X2=Z2/2ª
10. Imprimimos X1 y X2
11. Fin



**PRUEBAS DE ESCRITORIO:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Iteración** | **a** | **b** | **c** | **Salida** |
| **1** | **2** | **6** | **1** | **X1= X2=** |
| **2** | **0** | **3** | **2** | **Volvemos al inicio pues a debe ser dif de 0** |
| **3** | **4** | **2** | **6** | **X1= X2=** |

**ÁREA DE UN CÍRCULO**

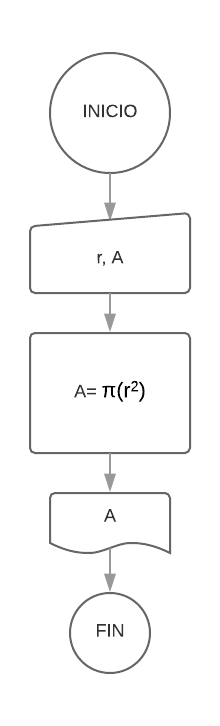
Algoritmo:

DATOS DE ENTRADA: Valor del Radio

Datos de salida: valor del Área

Restricciones: ninguna

1. Inicio
2. Conocer el radio
3. Operación:
4. Mostrar el área
5. Fin



**PRUEBA DE ESCRITORIO**

|  |  |  |
| --- | --- | --- |
| Iteración | Radio | Salida (Área) |
| 1 | 5 | 78.5398 |

**Fórmula**

Algoritmo:

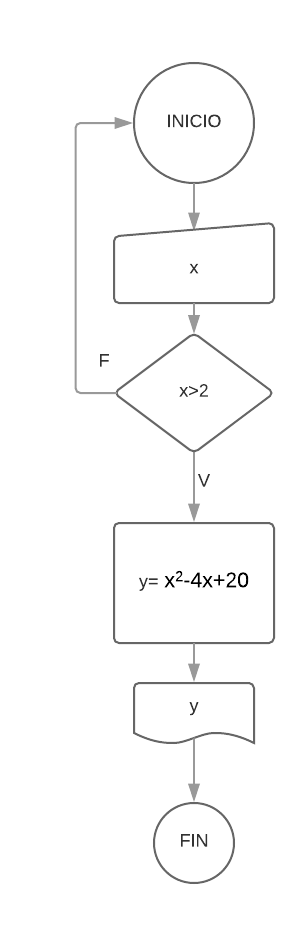
DATOS DE ENTRADA: Valor de X

DATOS DE SALIDA: Valor de Y

Dominio: (2,  ∞)

Restricciones: x>2

1. Inicio
2. Introducir el valor de “x”
3. Valuar si x>2
   1. Si es verdadero, ir al paso 4
   2. Si es falso, regresar al inicio.
4. Realizar la operación
5. Imprimir “y”
6. Fin



**Prueba de Escritorio:**

|  |  |  |
| --- | --- | --- |
| Iteración | Valor de “x” | Salida (“Y) |
| 1 | 3 | 17 |
| 2 | 1 | Volver al inicio |
| 3 | 2 | Volver al inicio |
| 4 | 4 | 20 |

**Fórmula**

Algoritmo:

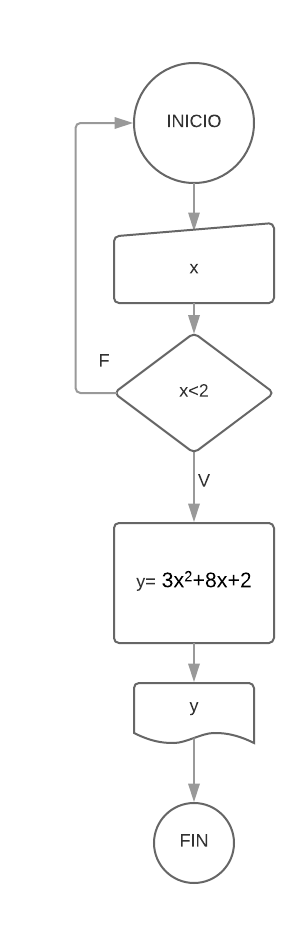
DATOS DE ENTRADA: Valor de X

DATOS DE SALIDA: Valor de Y

Dominio: (-∞,2)

Restricciones: x<2

1. Inicio
2. Introducir el valor de “x”
3. Valuar si x<2
   1. Si es verdadero, ir al paso 4
   2. Si es falso, regresar al inicio
4. Realizar la operación
5. Imprimir “y”
6. Fin



**Prueba de Escritorio:**

|  |  |  |
| --- | --- | --- |
| Iteración | Valor de “x” | Salida (“y”) |
| **1** | **1** | **13** |
| **2** | **5** | Volver al inicio |
| **3** | **2** | Volver al inicio |
| **4** | **-3** | **5** |

**CONCLUSIONES:**

A la hora de resolver un problema es necesario entender la solución que se necesita dar al mismo, y la mayoría de las veces esta solución no es solo un paso, sino un proceso.

Para resolverlo, es necesario el análisis de las necesidades que se requieren para llegar al resultado, donde identificamos un conjunto de datos de entrada, un proceso u operación y un conjunto de datos de salida, o sea, nuestra solución.

Una manera de organizar todos los pasos, ideas, restricciones, de este proceso es mediante un algoritmo y su representación gráfica, el diagrama de flujo, los cuales son muy útiles para el diseño de la solución. Es importante reconocer la naturaleza de cada parte de nuestra solución, para así representarlo de una manera clara, concisa y eficiente, pues ese es el objetivo de un algoritmo.