**Análisis y Diseño**

**Ejercicio 18: Desarrolle el análisis y diseño de un algoritmo que permita obtener las raíces de una**

**ecuación de segundo grado. Además, utilice la estructura según para el análisis de la discriminante de la ecuación cuadrática. Obviamente codifique en Processing**

**Resultado:**

**ANÁLISIS DEL PROBLEMA:**

**Objetivos:** calcular las raíces de una ecuación cuadrática.

**Análisis:**

**Datos de entrada:**

Los coeficientes de la ecuación cuadrática de la forma ax^2 +bx +c = 0, donde a,b, y

c son coeficientes reales y a no puede ser igual a 0.

**Datos de salida:**

Las raíces de una ecuación cuadrática que pueden ser reales o complejas

dependiendo del discriminante.

**Proceso:**

**¿Quién realiza el proceso?:** puede ser una persona o un programa

**¿Cuál es el proceso que realiza?:**

**Inicio:** el algoritmo comienza solicitando o asignando a los coeficientes a,b, y c de

una ecuación cuadrática de la forma ax^2 + bx + c =0

**Cálculo de discriminante:** Se calcula utilizando la fórmula D = b^2 – 4ac.

**Análisis de discriminante y cálculo de raíces**

**Fin**

| **NOMBRE DE LA ENTIDAD QUE RESUELVE EL PROBLEMA:** PROGRAMA O PERSONA |
| --- |
| **Variables:** // definir las variables  //Inicio  **Inicio**  **Leer a, b, c**  **D &lt;- b^2 - 4\*a\*c**  **Según D Hacer**  **Caso D &gt; 0:**  **x1 &lt;- (-b + raíz\_cuadrada(D)) / (2\*a)**  **x2 &lt;- (-b - raíz\_cuadrada(D)) / (2\*a)**  **Escribir Raíces reales y distintas: x1 = x1, x2 =x2**  **Caso D = 0:**  **x &lt;- -b / (2\*a)**  **Escribir &quot;Raíz real doble: x =, x**  **Caso Contrario:**  **parteReal &lt;- -b / (2\*a)**  **parteImaginaria &lt;- raíz\_cuadrada(-D) / (2\*a)**  **Escribir &quot;Raíces complejas: x1 =, parteReal, +parteImaginaria i , x2 =, parteReal ,**  **parteImaginaria,**  **FinSegún**  **fin** |
| **NOMBRE DEL ALGORITMO:** Calculadora\_deraicesdeecuaciones\_cuadraticas  **PROCESO DEL ALGORITMO:**  **//Inicio**  **void setup() {**  **/**/ No necesitamos un lienzo gráfico para este ejemplo  **noLoop(); //** Evita que draw() se ejecute continuamente  **}**  **void draw() {**  **// Coeficientes de la ecuación cuadrática: ax^2 + bx + c = 0**  **float a = 1; // Ejemplo**  **float b = 5;**  **float c = 6;**  **float discriminante = b\*b - 4\*a\*c;**  **float x1, x2; // Raíces**  **if (discriminante &gt; 0) {**  **x1 = (-b + sqrt(discriminante)) / (2\*a);**  **x2 = (-b - sqrt(discriminante)) / (2\*a);**  **println(&quot;Raíces reales y distintas: x1 = + x1 + , x2 = + x2);**  **} else if (discriminante == 0) {**  **x1 = -b / (2\*a);**  **println(Raíz real doble: x = + x1);**  **} else { // D &lt; 0**  **float realPart = -b / (2\*a);**  **float imaginaryPart = sqrt(-discriminante) / (2\*a);**  **println(&quot;Raíces complejas: x1 = + realPart + + + imaginaryPart + i , x2 = + realPart + - + imaginaryPart + i);**  **}**  **}**  **Fin** |

